SIMATIC S5

S5-95F
Programmable controller

Manual

EWA 4NEB 812 6220-02

Edition 01

STEP®, SINEC® and SIMATIC® are registered trademarks of Siemens AG.

Copyright© Siemens AG 1996

Subject to change without prior notice.

The reproduction, transmission or use of this document or its contents is not permitted without express written authority. Offenders will be liable for damages. All rights, including rights created by patent grant or registration of a utility model or design, are reserved.

EWA 4NEB 812 6220-02

Introduction

S5-95F Failsafe Mini PLC 1

Design, Functions and Operation 2

Guidelines for Planning and Installation of the Product 3

Installing and Connecting the Basic System 4

Expansion of Basic System with External I/Os 5

Addressing 6

Introduction to STEP 5 7

STEP 5 Operations 8

Blocks and Their Functions 9

The Integral Real-Time Clock 10

Analog Value Processing 11

Interrupt Processing 12

Connecting the S5-95F to SINEC L1 and PROFIBUS 13

Testing the User Program and Storing It on the Memory Submodule 14

Error Diagnosis and Elimination 15

High-Availability Failsafe Control with S5-95F 16

Application 17

Rules for Failsafe Operation of an S5-95F 18

Appendices A/B/C D/E

Contents

			Page
Introduction			**xvii**
1	**S5-95F Failsafe Mini PLC**		**1 - 1**
1.1	Typical Applications for the S5-95F Programmable Controller		1 - 6
1.2	S5-95F Hardware		1 - 8
1.3	S5-95F Programming and Parameterization Software		1 - 9
2	**Design, Functions and Operation**		**2 - 1**
2.1	Basic System Design - without External I/Os		2 - 1
2.2	Basic System Design - with External I/Os		2 - 2
2.3	Internal Functions		2 - 4
2.3.1	Integral Real-Time Clock		2 - 4
2.3.2	Diagnostic Byte		2 - 4
2.3.3	Communication Byte		2 - 4
2.4	The Programmable Controllers' Principle of Operation		2 - 5
2.4.1	Functional Units		2 - 5
2.4.2	Mode of Operation of the External I/O Bus		2 - 8
2.5	Operating Instructions		2 - 11
2.5.1	Displays and Controls		2 - 11
2.5.2	Operating Modes		2 - 13
2.5.3	Operating Statuses of the S5-95F Following Power-Up		2 - 17
2.5.4	Performing an Overall Reset on the S5-95F		2 - 18
2.5.5	Function of the Backup Battery		2 - 19
2.5.6	Memory Submodules		2 - 20
3	**Guidelines for Planning and Installation of the Product**		**3 - 1**
3.1	Guidelines on the Safe Integration of the Product into its Environment		3 - 1
3.2	Installation of Programmable Controllers in Accordance with Principles of EMC		3 - 2
3.2.1	Overview of Possible Interference		3 - 2
3.2.2	Most Important Basic Rules for Ensuring EMC		3 - 4
3.3	Installation of Programmable Controllers for EMC		3 - 5
3.3.1	Basic Rules for Assembling and Grounding the Inactive Metal Parts		3 - 5
3.3.2	Example of Rack Mounting for EMC		3 - 6

		Page
3.4	Wiring of Programmable Controllers for EMC	3 - 7
3.4.1	Routing of Cables	3 - 7
3.4.2	Equipotential Bonding	3 - 9
3.4.3	Shielding of Cables and Lines	3 - 10
3.4.4	Special Measures for Interference-Free Operation	3 - 11
3.4.5	Filters for 24 V DC Power Supply Units	3 - 13
3.4.6	Checklist for the Electromagnetically Compatible Installation of Control Systems	3 - 14
4	**Installing and Connecting the Basic System**	**4 - 1**
4.1	Basic System	4 - 1
4.1.1	S5-95F Basic Unit	4 - 2
4.1.2	Power Supply for the S5-95F Basic Unit	4 - 4
4.1.3	Setting the Subunit Identifier and the Length of the Fiber Optic Cable	4 - 5
4.2	Using the Onboard I/Os	4 - 7
4.2.1	Using the Failsafe, Onboard Digital Inputs	4 - 9
4.2.2	Short-Circuit Test for Sensor Lines	4 - 11
4.2.3	Using the Failsafe Onboard Digital Outputs	4 - 18
4.3	Connecting Actuators to Failsafe Digital Outputs	4 - 21
4.4	Onboard Interrupt Inputs	4 - 23
4.5	Onboard Counter Inputs for Counting and Monitoring Rotational Speed	4 - 23
4.5.1	Connection of Counter Inputs	4 - 24
4.5.2	Parameterizing Counter Inputs	4 - 25
4.5.3	Using Counter Inputs for Counting Tasks	4 - 26
4.5.4	Using Counter Inputs for Frequency Monitoring and Speed Monitoring	4 - 27
4.5.5	Scanning and Resetting the Counter Status	4 - 29
4.6	Failure and Monitoring of the Supply Voltages	4 - 30
4.7	Connector Pin Assignment of the Onboard I/Os	4 - 31
5	**Expansion of Basic System with External I/Os**	**5 - 1**
5.1	Assembling a Tier	5 - 1
5.2	Multi-Tier Expansion	5 - 5
5.3	Cabinet Mounting	5 - 6
5.3.1	Horizontal Mounting	5 - 7
5.3.2	Vertical Mounting	5 - 8
5.4	Connection Methods: Screw-Type Terminals and Crimp Snap-In	5 - 9

Page

5.5 Connection of Failsafe I/O Modules . . . . . 5 - 12
5.5.1 Expansion of Basic Unit with Failsafe Digital Input Module . . . . . 5 - 13
5.5.2 Expansion of Basic Unit with Failsafe Digital Output Module . . . . . 5 - 15
5.5.3 Expansion of Basic Unit with Failsafe Analog Value Processing . . . . . 5 - 21

5.6 Connection of Non-Failsafe I/O Modules . . . . . 5 - 23

5.7 Power Supplies for the S5-95F . . . . . 5 - 27

5.8 Electrical Potentials for the Onboard and External I/Os . . . . . 5 - 28
5.8.1 Onboard I/Os of the S5-95F Programmable Controller . . . . . 5 - 28
5.8.2 Potential Bonding and Galvanic Isolation of External I/Os . . . . . 5 - 29

5.9 Grounded or Non-Grounded S5-95F Configuration . . . . . 5 - 31

6 Adressing . . . . . 6 - 1

6.1 Address Assignments for Onboard I/Os . . . . . 6 - 2

6.2 Slot Numbering and Address Assignment for External I/Os . . . . . 6 - 2

6.3 Digital Modules . . . . . 6 - 4

6.4 Analog Modules . . . . . 6 - 5

6.5 Combined Input and Output Modules . . . . . 6 - 6
6.5.1 Digital Input/Output Module, 16 DI/16 DQ for 24 V DC . . . . . 6 - 6
6.5.2 Function Modules . . . . . 6 - 7

6.6 Structure of Process Input and Output Images . . . . . 6 - 8
6.6.1 Accessing the Process Input Image . . . . . 6 - 10
6.6.2 Accessing the Process Output Image . . . . . 6 - 11
6.6.3 Direct Access to Onboard I/Os . . . . . 6 - 12
6.6.4 Direct Access to External I/Os . . . . . 6 - 13

6.7 Interrupt Process Image Tables and Time-Controlled Program Processing in OB13 . . . . . 6 - 13
6.7.1 Accessing the Interrupt PII . . . . . 6 - 13
6.7.2 Accessing the Interrupt PIQ . . . . . 6 - 14

6.8 Address Assignments in RAM . . . . . 6 - 15

7 Introduction to STEP 5 . . . . . 7 - 1

7.1 Writing a Program . . . . . 7 - 1
7.1.1 Methods of Representation . . . . . 7 - 1
7.1.2 Operand Areas . . . . . 7 - 3
7.1.3 Circuit Diagram Conversion . . . . . 7 - 3

7.2 Program Structure . . . . . 7 - 4
7.2.1 Linear Programming . . . . . 7 - 4
7.2.2 Structured Programming . . . . . 7 - 5

Page

7.3 Block Types . . . . . 7 - 7
7.3.1 Organization Blocks (OBs) . . . . . 7 - 9
7.3.2 Program Blocks (PsB) . . . . . 7 - 11
7.3.3 Sequence Blocks (SBs) . . . . . 7 - 11
7.3.4 Function Blocks (FBs) . . . . . 7 - 11
7.3.5 Data Blocks (DBs) . . . . . 7 - 16

7.4 Program Processing . . . . . 7 - 18
7.4.1 START-UP Program Processing . . . . . 7 - 19
7.4.2 Cyclical Program Processing . . . . . 7 - 21
7.4.3 Maximum Response Time with Cyclical Program Processing . . . . . 7 - 23
7.4.4 Time-Controlled Program Processing . . . . . 7 - 25
7.4.5 Maximum Response Time with Time-Controlled Program Processing . . . . . 7 - 28
7.4.6 Interrupt-Driven Program Processing . . . . . 7 - 29

7.5 Processing Blocks . . . . . 7 - 30
7.5.1 Changing Programs . . . . . 7 - 30
7.5.2 Changing Blocks . . . . . 7 - 30
7.5.3 Compressing the Program Memory . . . . . 7 - 30

7.6 Number Representation . . . . . 7 - 31

8 STEP 5 Operations 5-Operationen . . . . . 8 - 1

8.1 Basic Operations . . . . . 8 - 1
8.1.1 Boolean Logic Operations . . . . . 8 - 2
8.1.2 Set/Reset Operations . . . . . 8 - 7
8.1.3 Load and Transfer Operations . . . . . 8 - 10
8.1.4 Timer Operations . . . . . 8 - 15
8.1.5 Counter Operations . . . . . 8 - 25
8.1.6 Comparison Operations . . . . . 8 - 29
8.1.7 Arithmetic Operations . . . . . 8 - 30
8.1.8 Block Call Operations . . . . . 8 - 32
8.1.9 Other Operations . . . . . 8 - 37

8.2 Supplementary Operations . . . . . 8 - 38
8.2.1 Load Operation . . . . . 8 - 39
8.2.2 Enable Operation . . . . . 8 - 40
8.2.3 Bit Test Operations . . . . . 8 - 41
8.2.4 Digital Logic Operations . . . . . 8 - 43
8.2.5 Shift Operations . . . . . 8 - 47
8.2.6 Conversion Operations . . . . . 8 - 49
8.2.7 Decrement/Increment . . . . . 8 - 51
8.2.8 Disabling/Enabling Interrupts . . . . . 8 - 52
8.2.9 "DO" Operation . . . . . 8 - 53
8.2.10 Jump Operations . . . . . 8 - 55
8.2.11 Substitution Operations . . . . . 8 - 57

Page

8.3 System Operations . . . . . 8 - 63
8.3.1 Set Operations . . . . . 8 - 63
8.3.2 Load and Transfer Operations . . . . . 8 - 64
8.3.3 Arithmetic Operations . . . . . 8 - 66
8.3.4 Other Operations . . . . . 8 - 67

8.4 Condition Code Generation . . . . . 8 - 68

8.5 Sample Programs . . . . . 8 - 70
8.5.1 Momentary-Contact Relay/Edge Evaluation . . . . . 8 - 70
8.5.2 Binary Scaler/Binary Divider . . . . . 8 - 70
8.5.3 Clock/Clock-Pulse Generator . . . . . 8 - 72

9 Blocks and Their Functions . . . . . 9 - 1

9.1 Organization Blocks . . . . . 9 - 1
9.1.1 Scan Time Trigger (OB31) . . . . . 9 - 2
9.1.2 Procedure after Battery Failure (OB34) . . . . . 9 - 2
9.1.3 Error Handling (OB37) . . . . . 9 - 2
9.1.4 PID Algorithm (OB 251) . . . . . 9 - 3

9.2 Integrated Function Blocks . . . . . 9 - 14
9.2.1 2-out-of-3 Evaluation for Failsafe Digital Inputs -FB234- . . . . . 9 - 15
9.2.2 Interfacing Operator Panels and Text Displays via the CP 521 SI's Serial Port -FB235- . . . . . 9 - 23
9.2.3 Function Block FB236 for Non-Equivalence and Watchdog Timers . . . . . 9 - 34
9.2.4 Function Block FB 237, Interpolation Block . . . . . 9 - 41
9.2.5 Code Converter : B4 - FB240 - . . . . . 9 - 53
9.2.6 Code Converter : 16 - FB241 - . . . . . 9 - 53
9.2.7 Multiplier : 16 - FB242 - . . . . . 9 - 54
9.2.8 Divider : 16 - FB243 - . . . . . 9 - 54
9.2.9 Additional Calling Up of Test Routines - FB252 - . . . . . 9 - 55
9.2.10 Depassivation Block - FB255 - . . . . . 9 - 61

9.3 Parameterizing Internal Functions in DB1 . . . . . 9 - 63
9.3.1 Configuration and Default Settings for DB1 . . . . . 9 - 64
9.3.2 How to Assign Parameters in DB1 without COM 95F . . . . . 9 - 66
9.3.3 Rules for Setting Parameters in DB1 . . . . . 9 - 66
9.3.4 How to Recognize and Correct Parameter Errors . . . . . 9 - 68
9.3.5 Transferring Changed DB1 Parameters to the S5-95F Programmable Controller . . . . . 9 - 69
9.3.6 Reference Guide for Setting Parameters in DB1 . . . . . 9 - 70

10 The Integral Real-Time Clock 10- 1

10.1 Operating Principle and Parameterization of the Integral Real-Time Clock 10- 1
10.1.1 Setting the Clock Parameters in DB1 . . . . . 10- 1
10.1.2 Operating Principle of the Clock 10- 1

		Page
10.1.3	Transfer and Battery Backup of Clock Parameters	10- 2
10.1.4	Entering the Clock Time Correction Factor	10- 2
10.2	Structure of the Clock Data Area	10- 3
10.3	Structure of the Status Word and How to Scan it	10- 6
10.4	Using the Programmer to Read and Set the Integral Real-Time Clock	10- 9
10.5	Programming the Integral Real-Time Clock in the Control Program	10- 10
10.5.1	Reading and Setting the Clock	10- 10
10.5.2	Storing the Updated Time/Date after a RUN to STOP or RUN to POWER OFF Transition	10- 14
10.5.3	Setting the Prompt Time	10- 14
10.5.4	Setting the Operating Hours Counter	10- 18
11	**Analog Value Processing**	**11- 1**
11.1	Analog Input Modules (Type P)	11- 1
11.2	Connecting Current and Voltage Sensors to Analog Input Modules	11- 1
11.2.1	Voltage Measuring with Isolated/Non-Isolated Thermocouples	11- 2
11.2.2	Two-Wire Connection of Voltage Sensors	11- 3
11.2.3	Two-Wire Connection of Current Sensors	11- 4
11.2.4	Connection of Two-Wire and Four-Wire Transducers	11- 5
11.3	Start-Up of Analog Input Modules	11- 7
11.4	Analog Value Representation of Analog Input Modules	11- 8
11.5	Analog Output Modules (Type W)	11- 10
11.5.1	Connection of Loads to Analog Output Modules	11- 10
11.5.2	Analog Value Representation of Analog Output Modules	11- 11
11.6	Analog Value Conversion: Function Blocks FB250 and FB251	11- 13
11.6.1	Reading in and Scaling an Analog Value - FB250 -	11- 13
11.6.2	Output of Analog Value - FB251 -	11- 16
11.7	Failsafe Analog Value Processing - FB232 and FB233 -	11- 18
11.7.1	Discrepancy Analysis for Two Analog Inputs - FB232 -	11- 19
11.7.2	Parameterization Example for FB232	11- 27
11.7.3	Discrepancy Analysis for Three Analog Inputs - FB233 -	11- 28
11.7.4	Combining FB232 and FB233	11- 39
11.8	Circuit Versions for Function Blocks FB232 and FB233	11- 39
11.8.1	Circuit Type R4.2 for AK4	11- 42
11.8.2	Circuit Type R4.4 for AK4	11- 43
11.8.3	Circuit Type R5.1 for AK5	11- 44
11.8.4	Circuit Type R6.1 for AK6	11- 45

Page

11.8.5 Circuit Type R6.2 for AK6 . . . . . 11- 47
11.8.6 Circuit Type R6.3 for AK6 . . . . . 11- 49
11.8.7 Circuit Type R6.4 for AK6 . . . . . 11- 51
11.8.8 Circuit Type R6.5 for AK6 . . . . . 11- 53

12 Interrupt Processing 12- 1

12.1 Using Onboard Interrupt Inputs 12- 1

12.2 Asynchronous Interrupt Processing in OB2 . . . . . 12- 2
12.2.1 Programming OB2 . . . . . 12- 4
12.2.2 Programming Interrupt Responses in OB2 . . . . . 12- 6
12.2.3 Connecting Interrupt Inputs 12- 7

12.3 Synchronous Interrupt Processing in OB3 . . . . . 12- 9
12.3.1 Programming OB3 . . . . . 12- 12
12.3.2 Programming Interrupt Responses in OB3 . . . . . 12- 13

12.4 Interrupt Response Times for the S5-95F 12- 14

13 Connecting the S5-95F to SINEC L1 and PROFIBUS 13- 1

13.1 Options for Connecting the S5-95F to the SINEC L1 LAN 13- 1

13.2 Non-Failsafe Data Interchange over SINEC L1 . . . . . 13- 3
13.2.1 Initializing the S5-95F for Non-Failsafe Data Interchange 13- 4
13.2.2 Coordinating Non-Failsafe Data Interchange in the User Program 13- 6
13.2.3 Transmitting Non-Failsafe Data 13- 7
13.2.4 Receiving Non-Failsafe Data 13- 8

13.3 Failsafe Data Interchange over SINEC L1 . . . . . 13- 10
13.3.1 Initializing the S5-95F for Failsafe Data Interchange 13- 12
13.3.2 Coordinating Failsafe Data Interchange in the User Program 13- 15
13.3.3 Sending Failsafe Data 13- 18
13.3.4 Receiving Failsafe Data 13- 20

13.4 SINEC L1 Safety Times 13- 25
13.4.1 SINEC L1 Safety Time for Receive 13- 25
13.4.2 SINEC L1 Safety Time for Send 13- 28
13.4.3 Load Placed on the System by the SINEC L1 Channel 13- 29
13.4.4 Response Time During SINEC L1 Traffic 13- 30
13.4.5 Transmission of Error Messages to the SINEC L1 Master 13- 31
13.4.6 Example: SINEC L1 Cycle Time and SINEC L1 Safety Time 13- 31

13.5 Connecting the S5-95F to the PROFIBUS 13- 34

Page

14 Testing the User Program and Storing It on the Memory Submodule 14- 1

14.1 Testing and Debugging the User Program 14- 1
14.1.1 Program-Dependent Signal Status Display "STATUS" 14- 1
14.1.2 Direct Signal Status Display "STATUS VAR" 14- 3
14.1.3 Forcing Variables with "FORCE VAR" 14- 4
14.1.4 Search Function 14- 4

14.2 Interrupt Analysis with the Programmer 14- 5
14.2.1 The "ISTACK" Analysis Function 14- 5
14.2.2 Descriptions of the ISTACK Flags 14- 9

14.3 Program Errors 14- 10
14.3.1 Determining the Error Address 14- 10
14.3.2 Program Trace with the "BSTACK" Function 14- 12

14.4 Programmer Control Functions 14- 13

14.5 Measures for Securing the User Program Against Errors 14- 15

14.6 Storing the User Program on Memory Submodules 14- 16

15 Error Diagnosis and Elimination 15- 1

15.1 S5-95F Responses to Errors 15- 1
15.1.1 Hard STOP 15- 1
15.1.2 Soft STOP 15- 1
15.1.3 Passivation of a Signal Group 15- 1
15.1.4 Reaction in the User Program 15- 1
15.1.5 Error Indication 15- 2
15.1.6 System Event DB and OB37 ..... 15- 2

15.2 Error Indicator on the S5-95F Basic Unit 15- 2

15.3 System Event Data Block DB254 ..... 15- 3
15.3.1 System Identification and ID Number Entries 15- 3
15.3.2 Standard FB and Signature 15- 4
15.3.3 Information on System Responses 15- 4
15.3.4 Image of the Signal Groups 15- 5
15.3.5 Image of the Error Groups 15- 5
15.3.6 Static Image of I/O Errors 15- 6
15.3.7 Image of SINEC L1 Errors 15- 7
15.3.8 Error Stack Entries 15- 8
15.3.9 Evaluating the Error Block 15- 10

15.4 Acknowledging Errors and Deleting Entries in the System Event DB 15- 19

15.5 Printer Output of Error Messages Via a CP 521 SI 15- 20

		Page
15.6	Forwarding Error Messages to the SINEC L1 Master	15- 20
15.7	Evaluating Cycle Time Statistics	15- 21
15.8	Diagnostic Byte for Battery and Load Voltage	15- 22
16	**High-Availability Failsafe Control with S5-95F**	**16- 1**
16.1	Schematic Wiring Diagram of the Data Link	16- 2
16.2	Data Interchange for the High-Availability S5-95F -FB 230-	16- 4
16.2.1	Block Parameters for FB 230	16- 4
16.2.2	Description of the Parameters for FB 230	16- 5
16.2.3	Operands Reserved by FB 230	16- 10
16.2.4	How FB 230 Works	16- 11
16.2.5	Operating States and Transitional States of the High-Availability S5-95F	16- 14
16.2.6	Time Response of the FB 230	16- 16
16.3	Initializing Parameters with COM 95F	16- 17
16.4	What to Watch for When Writing User Programs	16- 17
16.4.1	Structure of a Program for the High-availability S5-95F	16- 18
16.5	Timing in the User Program	16- 20
16.6	Using Analog Inputs in the H/F System	16- 21
16.7	Using SINEC L1 in the H/F System	16- 21
16.8	Sample Program for Initializing FB 230	16- 22
16.9	Connecting I/Os to High-Availability S5-95Fs	16- 27
16.9.1	Connecting Digital Inputs	16- 27
16.9.2	Connecting Digital Outputs	16- 30
16.9.3	Connecting Analog Inputs	16- 34
16.10	Timing for H/F Systems -FB 231 and FB 238-	16- 36
16.10.1	Block Parameters for FB 231	16- 36
16.10.2	Block Parameters for FB 238	16- 39
17	**Application**	**17- 1**
17.1	Hardware Prerequisites	17- 1
17.2	Process Description	17- 2
17.3	Installation and Wiring	17- 3
17.4	Entering the User Program	17- 5
17.5	Entering Configuring Data with COM 95F	17- 8
17.6	Testing and Error Simulation	17- 11

		Page
18	**Rules for Failsafe Operation of an S5-95F**	**18- 1**
18.1	Acceptance Test for a System Containing an S5-95F	18- 1
18.1.1	Planning Phase	18- 2
18.1.2	Pre-Acceptance Inspection	18- 4
18.1.3	Acceptance Test	18- 5
18.2	I/Os	18- 7
18.2.1	Circuit Diagram for I/Os	18- 7
18.2.2	Discrepancy Times	18- 15
18.3	Operating Modes	18- 17
18.4	Entering the System Identification and ID Number in the System Event DB	18- 17
18.5	Memory Submodules for Safety Mode	18- 18
18.6	Function of the Backup Battery	18- 18
18.7	Retentivity of Timers, Counters and Flags	18- 19
18.8	Requirements for Sensors for Failsafe Digital Input Modules	18- 19
18.9	Requirements for Actuators for Failsafe Digital Output Modules	18- 20
18.10	Response to I/O Errors	18- 21
18.10.1	Passivating of I/Os	18- 22
18.10.2	Depassivating I/Os	18- 23
18.10.3	Standard Value Formation and Reaction at the User Level	18- 25
18.11	Repairs	18- 26
18.12	S5-95F Response Times	18- 27
18.12.1	Response Time and Signal Duration for Cyclical Program Processing	18- 27
18.12.2	Response Time and Minimum Signal Duration for Time-Controlled Program Processing	18- 28
18.12.3	Response Times and Minimum Signal Duration for OB2 Interrupts	18- 29
18.12.4	Response Times and Minimum Signal Duration for OB3 Interrupts	18- 30
18.13	Special Programming Features	18- 31
18.13.1	Disabling/Enabling Interrupts	18- 31
18.13.2	Restrictions on LIR, TIR, TNB and TBS Operations	18- 32
18.13.3	STOP Operation in the User Program	18- 32
18.13.4	Waiting Times in the User Program	18- 33
18.13.5	Scratchpad	18- 33
18.13.6	Post-Loading STEP 5 Blocks in Test Mode	18- 33
18.13.7	Monitoring the Supply Voltage for OB3 Interrupt DI	18- 33

Page

18.13.8 Trigger Edge for OB3 Interrupts 18- 34
18.13.9 Measures for Securing the User Program Against Errors 18- 34

18.14 Addressing and Address Assignments 18- 35
18.14.1 Address Assignments for Onboard I/Os 18- 36
18.14.2 Slot Numbering and Address Assignment for External I/Os 18- 36

18.15 Loadable and Integral Function Blocks 18- 38
18.15.1 Loadable Function Blocks 18- 38
18.15.2 Integral Function Blocks 18- 39

18.16 Programmer-Based Operator Input Functions 18- 41

18.17 Connecting Operator Panels and Text Displays 18- 43

18.18 SINEC L1 LAN 18- 44

18.19 SINEC L1 Safety Times 18- 46
18.19.1 SINEC L1 Safety Time for Receive 18- 46
18.19.2 SINEC L1 Safety Time for Send 18- 49
18.19.3 Response Time During SINEC L1 Traffic 18- 50

18.20 Filter for 24 V DC Power Supply Units 18- 50

18.21 EMC of the IM 316-MA12 . . . . . 18- 51

Appendices

A Module Spectrum . . . . . A - 1

A.1 General Technical Specifications for Failsafe Modules . . . . . A - 4
A.1.1 Programmable Controller . . . . . A - 5
A.1.2 Digital Modules . . . . . A - 7
A.1.3 Analog Modules . . . . . A - 9
A.1.4 Bus Units . . . . . A - 11
A.1.5 Interface Modules . . . . . A - 13

A.2 Standard Modules from the S5-100U Range . . . . . A - 14
A.2.1 Digital Input Modules . . . . . A - 16
A.2.2 Digital Output Modules . . . . . A - 23
A.2.3 Digital Input/Output Modules . . . . . A - 31
A.2.4 Analog Input Modules . . . . . A - 33
A.2.5 Analog Output Modules . . . . . A - 66
A.2.6 Function Modules . . . . . A - 75
A.2.7 Bus Units . . . . . A -106
A.2.8 Interface Modules . . . . . A -108

			Page
B	**Dimension Drawings**		**B - 1**
	B.1	S5-95F Programmable Controller	B - 1
	B.2	Bus Units	B - 2
	B.3	Interface Modules	B - 4
	B.4	Standard Mounting Rails	B - 5
C	**Operations List**		**C - 1**
	C.1	Operations List	C - 1
	C.1.1	Basic Operations	C - 1
	C.1.2	Supplementary Operations	C - 7
	C.1.3	System Operations	C - 12
	C.1.4	Evaluation of CC 1 and CC 0	C - 12
	C.2	Machine Code Listing	C - 13
	C.3	List of Abbreviations	C - 16
D	**Guidelines for Handling Electrostatic Sensitive Devices (ESD)**		**D - 1**
E	**Prototype Test Certification**		**E - 1**
Index			

Introduction

Please read the introduction carefully. You will then find it easier to use the manual and this will save time.

The S5-95F is a failsafe programmable controller for the lower and mid performance ranges. It is for use wherever safety is the first priority and where potential dangers must be avoided. In order to use the controller to its best advantage you require detailed information.

In this manual we have attempted to present the necessary information as completely and as well organized as possible. Certain information is repeated in various chapters so that you do not have to leaf through the manual to find what you need.

On the following pages of this introduction you will find information that will make it easier for you to use the manual. This section explains how the manual is organized.

Contents

- Hardware Description (Chapters 1, 2, 3)
  These chapters describe the controller: how it fits into the SIMATIC S5 family of programmable controllers, how it functions, and how you install and connect it.

- Startup Information (Chapters 4, 5, 6)
  These chapters summarize the information you need to start up your programmable controller.

- The Programming Language and Blocks for the S5-95F (Chapters 7, 8, 9)
  These chapters describe the structure, operations, and structuring aids of the STEP 5 programming language and the function of the integral blocks.

- Functions of the S5-95F (Chapters 10, 11, 12, 13)
  Each of these chapters contains a complete description of a particular function, from wiring to programming (subjects include: integral clock, analog value processing, interrupt processing, SINEC L1).

- Diagnostics and Troubleshooting (Chapters 14, 15)
  These chapters describe the diagnostics and test means available to remedy hardware and software faults.

- High-availability Failsafe Controller with S5-95F (Chapter 16)
  This chapter describes the hardware configuration and the parameterization of the integral function block for linking two S5-95Fs.

- Example (Chapter 17)
  A simple example illustrates how easy it is to set up and initialize an S5-95F control system.

- Rules for Failsafe Applications (Chapter 18)
  This chapter is a brief repetition of the above topics. It is also intended to serve as a reference and to simplify system acceptance.

- Module Spectrum (Appendix A)
  This chapter contains the technical specifications of the S5-95F and information on the modules that you can use to expand your controller.

- Overviews (Appendices B through D)
  In these chapters you will find not only dimension drawings and a complete list of operations but also guidelines for handling electrostatic sensitive devices (ESD Guidelines).

- Prototype Test Certification (Appendix E)
  Certificates on prototype tests can be obtained from us on request.
  Please direct inquiries to the address specified in Appendix E.

You will find correction pages at the end of the manual. Use them to indicate any corrections, additions or suggestions for improvement you might have. Send these suggestions to us. They will help us to improve the next edition of this manual.

Conventions

This system manual is organized in menu form to make it easier for you to find information. This means the following:

- Each chapter is marked with printed tabs.
- At the front of the system manual is an overview page that lists the title of each chapter.
  Following this page, you will find a table of contents.
- At the beginning of each chapter is a table of contents for that chapter. Each chapter has three level headings that are numbered. The fourth level heading is not numbered but appears in **boldface type**.
- Pages, figures, and tables are numbered separately for each chapter. On the back of the table of contents for each chapter you will find a list of the figures and tables that appear in that chapter.

This system manual employs the following specific structuring devices:

- Specific terms have characteristic abbreviations (e.g. programmer is PG).
  Appendix A contains a list of abbreviations.
- Footnotes are marked with a raised number (e.g."¹"), or a raised asterisk (" *") . You will find the corresponding explanations in the lower margin of the page or under a figure or table if the footnotes appears in one of these.
  Lists are designed with bullets (• as in this particular listing) or with hyphens (-).
  Procedures are marked with black triangles ( ) and must be performed in the sequence presented.
- Cross references are indicated as follows:
  (see section 7.3.2)
  There are no references to specific page numbers.
- Dimensions in drawings are indicated in millimeters and inches.

Especially important information appears in framed boxes such as the following:

> **Warning**

You will find definitions for the terms "Note", "Safety Note", "Caution", and "Warning" in the Safety-Related Guildelines for the User at the end of the introduction.

Scope of this Manual

This manual contains a detailed description of our S5-95F system with basic units 6ES5 095-8FB01, version 1. The necessary COM 95F parameter assignment software is described in a separate manual.

The predecessor units 6ES5 095-8FAxx are described in the S5-95F manual with the Order No. 6ES5 998-1MFxx. Also please observe the additional product information.

Compatibility of basic units 6ES5 095-8FB01 and 6ES5 095-8FAxx

6ES5 095-8FB01 basic units are compatible to predecessor units 6ES5 095-8FAxx. Please note, however, that systems consisting of S5-95Fs must always have two identical basic units (the two basic units must have the same Order No. and the same revision level).

Changes with Regard to Programmable Controller 6ES5 095-8FAxx

Please note that the functionality of basic units 6ES5 095-8FB01 is more extensive that that of the preceding units.

The principle differences between an S5-95F programmable controller with basic units 6ES5 095-8FB01 and its precedessor are as follows:

- 2-out-of-3 evaluation for failsafe digital inputs (see Chapter 9.2.1)
- Integrated function blocks for connecting operator panels and text displays with recording function in failsafe mode (see Chapter 9.2.2)
- New (subunit-specific) short-circuit test for lines to failsafe digital inputs (see Chapter 4.2.2)
- Failsafe analog value processing (see Chapter 11.7)
- Establishment of high-availability failsafe control via DI/DQ link (see Chapter 16)
- Shorter execution and response times through the use of a new processor (see Appendix C)
- Shorter off-times when testing the failsafe digital outputs (see Chapter 4.3)
- Failsafe speed monitoring (see Chapter 4.5)

Training

SIEMENS offers a wide range of training courses for SIMATIC S5 users. Contact your Siemens representative for more information.

Safety-Related Guildelines for the User

This document provides the information required for the intended use of the particular product. The documentation is written for technically qualified personnel.
Qualified personnel as referred to in the safety guidelines in this document as well as on the product itself are defined as follows:

- System planning and design engineers who are familiar with the safety concepts of automation equipment.
- Operating personnel who have been trained to work with automation equipment and are conversant with the contents of the document in as far as it is connected with the actual operation of the plant.
- Commissioning and service personnel who are trained to repair such automation equipment and who are authorized to energize, de-energize, clear, ground, and tag circuits, equipment and systems in accordance with established safety practice.

Danger Notices

The notices and guidelines that follow are intended to ensure personal safety, as well as protect the products and connected equipment against damage.

The safety notices and warnings for protection against loss of life (the users or service personnel) or for protection against damage to property are highlighted in this document by the terms and pictograms defined here. The terms used in this document and marked on the equipment itself have the following significance:

Note

contains important information about the product, its operation or a part of the document to which special attention is drawn.

Safety Note

contains important information relevant for the acceptance and safety-related use of the product.

Caution

indicates that minor personal injury or property damage can result if proper precautions are not taken.

Warning

indicates that death, severe personal injury or substantial property damage can result if proper precautions are not taken.

Proper Usage



Warning

- The equipment/system or the system components may only be used for the applications described in the catalog or the technical description, and only in combination with the equipment, components, and devices of other manufacturers as far as this is recommended or permitted by Siemens.
- The product will function correctly and safely only if it is transported, stored, set up, and installed as intended, and operated and maintained with care.

1	S5-95F Failsafe Mini PLC	
1.1	Typical Applications for the S5-95F Programmable Controller	1 - 6
1.2	S5-95F Hardware	1 - 8
1.3	S5-95F Programming and Parameterization Software	1 - 9

Figures

1-1 Risk Graph and Quality Levels . . . . . 1 - 2
1-2 Burner Control of a Power Plant . . . . . 1 - 4
1-3 Press Control . . . . . 1 - 5
1-4 Basic System . . . . . 1 - 8

Tables

1-1 Comparison between Quality Levels to DIN V 19250 and IEC 65A (Sec) 123 . . . . . 1 - 2
1-2 Typical Applications for the S5-95F . . . . . 1 - 6
1-3 The Major Characteristics of the S5-95F and S5-115F at a Glance . . . . . 1 - 11

1 S5-95F Failsafe Mini PLC

The SIMATIC S5 system family offers programmable controllers in all performance classes, from the mini PLC to the high-end PLC.

For your failsafe-oriented applications you can choose between the modular S5-115F and the compact S5-95F. Both have been prototype-tested by the Bavarian Technical Inspectorate (TUEV Bayern), and both meet a number of safety standards which are listed in the prototype test certificate.

Additional tests and certifications are available for the S5-95F from the Institute for Industrial Safety of the Statutory Industrial Accident Insurance (BIA), the Swiss Institute for Industrial Safety (SUVA), the Professional/Trade Association (BG), the Federal Bureau of Railways (EBA), and from Underwriters Laboratories (UL). A summary of applications and their safety regulations is presented in Chapter 1.1.

A copy of the prototype test certificate as well as copies of the certificates of approval can be obtained from us on request (see Appendix E).

The following approvals/authorizations were granted for the S5-95F for failsafe applications:

- Safety classes 2 to 5 in conformity with the manual published by the German Technical Inspectorate
- Safety Integrity Levels 1 to 3 in conformity with IEC 65A (Sec) 123
- Quality levels 1 to 6 in conformity to DIN V 19250
- Quality levels for signalling systems of the German Federal Railways (Deutsche Bahn) in conformity with Mue 8004
- Quality levels in conformity with UL 1998

The quality levels to DIN V 19250 can be compared with the Safety Integrity Level in accordance with IEC 65A (Sec) 123, Fig. 7 of the normative Annex A (see Table 1-1).

TUEV Safety Classes

The German Technical Inspectorate (TUEV) has arranged the requirements laid down in the regulations into five safety classes, with the strictest requirements in class 1. It should be noted that the safety class is not necessarily an indication of the risk involved in the areas of application in question.

Integrity Level to IEC 65A (Sec) 123 (05.92)

Safety Integrity distinguishes between classes 1 to 4, with the lowest requirements in class 1. Safety Integrity can be defined as follows:
Safety Integrity is the likelihood of a safety-related system satisfactorily performing the required safety functions under all the stated conditions within a stated period of time.

The quality levels to DIN V 19250 and the Safety Integrity Levels can be compared as follows:

Table 1-1. Comparison Between Quality Levels to DIN V 19250 and IEC 65A (Sec) 123

Quality Level to DIN V 19250	System Integrity Level to IEC 65A (Sec) 123
2 and 3	1
4	2
5 and 6	3
7 and 8	4

Quality Levels to DIN V 19250

The classification to DIN 19250 is not based on existing regulations but rather uses risk parameters to define the risk inherent in a process. Risk parameters are the degree of damage, duration of stay in hazardous areas, the possibility of avoiding danger and the probability of undesired events arising. The risk graph gives information on the quality level in the case of protection facilities for measurement, open-loop control and closed-loop control.

DIN V 19250 classifies systems according to 8 quality levels. The quality levels of individual protection functions can be determined by means of a risk graph.

Each combination of risk parameters results in a "risk evaluation package". The different gradings permit a total of 48 combinations.

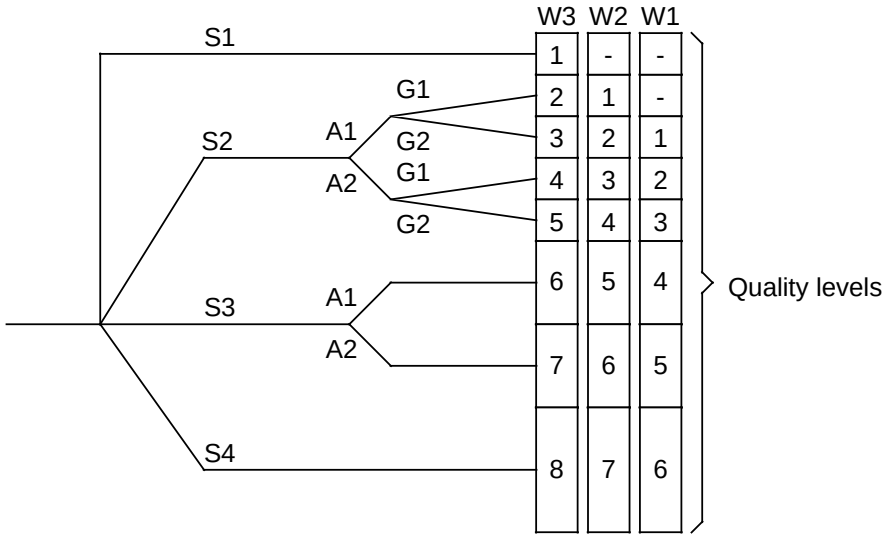


Figure 1-1. Risk Graph and Quality Levels

Risk Parameters

- Extent of damage

  S1: Minor injury or minor harm to the environment
  S2: Major irreversible injury of one or several persons or the death of one person or temporary major harm to the environment
  S3: Death of several persons or permanent major damage to the environment
  S4: Catastrophic effects, large number of deaths

- Duration of stay in hazardous area

  A1: Seldom to occasionally
  A2: Frequently to permanently

- Danger prevention

  G1: Possible under certain conditions
  G2: Hardly possible

- Probability of undesired events arising

  W1: Very low
  W2: Low
  W3: Relatively high

Example: Power plant burner control
The protection equipment of a power plant burner control must avoid gathering of an explosive gas-air mixture in the combustion chamber.

The undesired event is the ignition of the gas-air mixture resulting in the explosion of the firing plant.

The death of several persons must be reckoned with if persons stay in the hazardous area (S3).

In our case, we assume that persons stay only seldom in the hazardous area (A1).

Without protection equipment, the probability of the undesired event arising is relatively high (W3). According to the risk graph, this results in quality level 6.

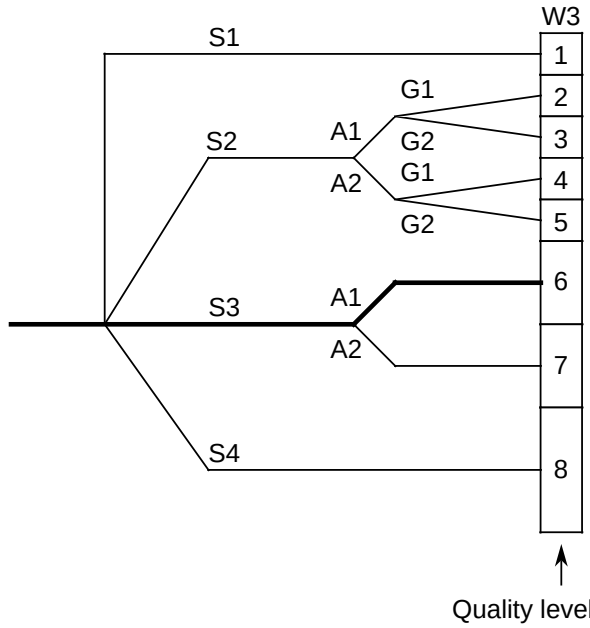


Figure 1-2. Burner Control of a Power Plant

Example: Press control
The protection equipment of a press control must avoid uncontrolled, dangerous movements.

The undesired event is the uncontrolled starting of the press from standstill or premature stopping of the press on request. The result of such an event can be major irreversible injury or the death of the operator (S2).

In this case, we assume that the operator stays frequently to permanently in the hazardous area (A2). Since press movements are fast processes, the danger can hardly be prevented (G2).

Without protection equipment, the probability of the undesired event arising is relatively high (W3). According to the risk graph, this results in quality level 5.

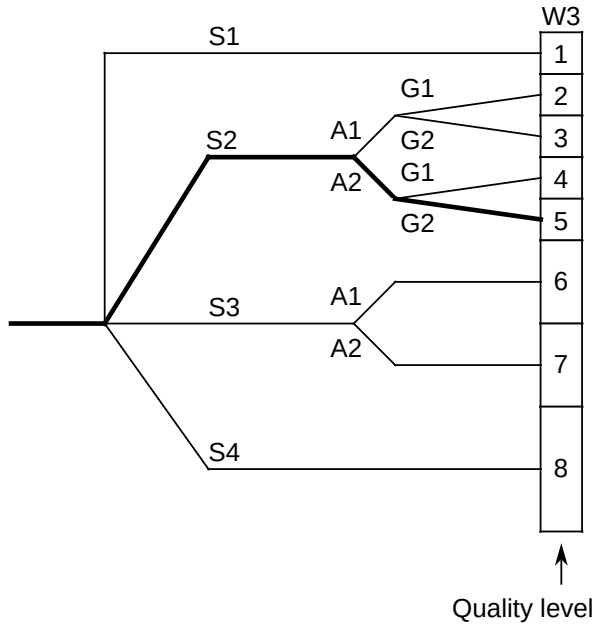


Figure 1-3. Press Control

1.1 Typical Applications for the S5-95F Programmable Controller

The following table contains applications of the S5-95F. It also lists the relevant national and international regulations and standards and the quality levels to DIN V 19250.

Table 1-2. Typical Applications for the S5-95F

Application	Primary Regulations	Additional Regulations		Quality Level
Mechanical presses	prEN 692	ZH 1/456	VBG 7/5.1	5
Hydraulic presses	prEN 693	ZH 1/457	VBG 7/5.2	5
Spindle presses	ZH 1/457	VBG 7/5.3		5
Stamping, bending, folding presses	ZH 1/387			5
Other industrial machines	IEC 204-1, EN 60204-1	DIN VDE 0113-1	EN 954	5
Protective-guard interlocking devices	EN 1088			5
Two-hand switching equipment	EN 574	DIN 24 980		1 to 5
Non-contact protective equipment (light curtains)	prEN 50100, Parts 1 and 2			2 to 5
EMERGENCY-OFF, EMERGENCY-STOP circuits	EN 418	EN 60947-5-1	EN 60204, Part 1	5
Gate drives	ZH 1/494	DIN VDE 0100	VDE 0700	3 to 5
Closing edge protective devices	ZH 1/494			2
Pressure-sensitive edges	DIN V 31006, Part 2			3 to 5
Printing and paper machines	prEN 1010	ZH 1/170	VBG 7/l	1 to 5
Fire alarm and extinguisher systems (only for processes with safe OFF position)	VDE 0833, Part 1 and 2	DIN 14 675		6
Burner controls	DIN VDE 0116, prEN 50156-1 EN 298	DIN VDE 0160,	VDE 0110	5
Pressure limiters	DIN 3398 (oil, gas)	VdTÜV Instruc-tions 100	AD Instruc-tions A6	6

Table 1-2. Typical Applications for the S5-95F (Continued)

Application	Primary Regulation	Additional Regulations		Quality Level
Gas high pressure pipes	TRGL 181	DIN VDE 0160	VDE 0800	6
Liquid pipes	TRBF 301			6
Automated transport systems	ZH 1/473			4
Passenger and load escalators	EN 81	TRA 200	TRA 101	6
Underground mining plants (without explosion protection)	DIN VDE 0118	DIN 57 118		6
Mining conveyance systems				5
Moving staircases, passenger conveyors	DIN EN 115			6
Belt conveyors	ZH 1/261			6
Lifting platforms	VBG 14			5
Cable railways	BO Seil	DIN VDE 0116		3 to 6
Signalling systems of the German Federal Railway (Deutsche Bahn)	Mü 8004			Not applicable
Railway stations				4 to 6
Grade crossings				6
Railway signalling systems		VDE 0800	DIN VDE 0160	4 to 6
Block safety of railway systems				6
Road traffic signalling systems	DIN VDE 0832	VDE 0800	DIN VDE 0160	5
Process control equipment	VDI/VDE 2880 VDI/VDE 2180	VDI/VDE 3541 VDI/VDE 3542		4 to 6 4 to 6
Process furnaces				6
Disintegrators, centrifuges	VBG 7/Z			4
Overfill protection equipment	VdTÜV Instruc-tions 100	DIN VDE 0116 prEN 50156-1	DIN VDE 0160	5
Pollutant filters				5

1.2 S5-95F Hardware

The basic configuration of the S5-95F programmable controller comprises two basic units which are connected via a fiber optic link.

The basic configuration offers you the following onboard I/Os:

- 16 failsafe digital inputs
- 4 failsafe interrupt inputs
- 2 failsafe counter inputs (for counting and speed monitoring)
- 8 failsafe digital outputs
- 8 non-failsafe digital outputs (4 DQs for each subunit)

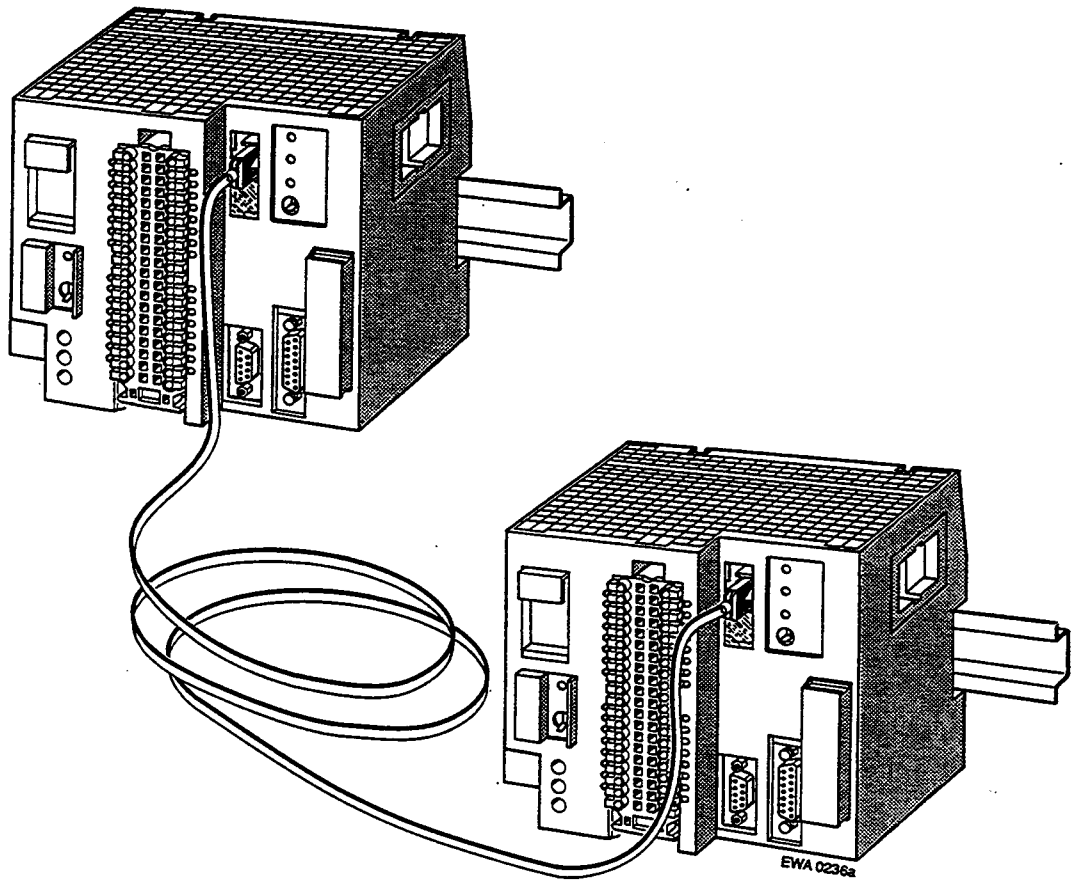


Figure 1-4. Basic System

If you require more than the available onboard I/Os, you can expand your basic system with additional external I/Os.

1.3 S5-95F Programming and Parameterization Software

Programming Instead of Wiring

Conventional controls using relays or contactors are hardwired. Their functions are implemented by wiring the switching elements. If the control task changes, time-consuming modifications of the wiring have to be made.

With the S5-95F, the sensors and actuators are connected to the programmable controller. The control task is written in a program where the only thing you do is to evaluate the signal states of the sensors and actuators connected. Modification of functions, testing and startup are thus simplified considerably.

STEP 5 Programming Language

This language enables the first-time user to become quickly familiar with PLC technology. All SIMATIC S5 PLCs, including the S5-95F, can be programmed using the STEP 5 language. STEP 5 is easy to comprehend and makes developing of programs user-friendly, flexible and economical.

STEP 5 offers three different methods of representation:

- Statement list (STL)
  Uses mnemonics for statements
- Control system flowchart (CSF)
  Uses logic representation
- Ladder diagram (LAD)
  Similar to a circuit diagram

Programming

Programming of the S5-95F means that the redundant structure is not taken into account (simulation of single-channel design).

This means that the 2-channel failsafe I/Os are addressed in the STEP 5 program only under one address.

Parameter Assignment

The man-machine interface (MMI) of the S5-95F is parameterized like a single-channel controller. The corresponding parameters are set in DB1 of the COM 95F parameterization software.

Parameter Assignment with COM 95F

The COM 95F parameterization software offers you a user-friendly, interactive operator interface. It simplifies

- Entering of system parameters
- Startup
- Error diagnostics
- On-site acceptance

Software Tools for Acceptance

A software-supported acceptance procedure has been developed to simplify system acceptance. It has been harmonized and agreed upon with the German Statutory Industrial Accident Insurance Institution (BG), the BG Institute for Industrial Safety (BIA) and the German Technical Inspectorate (TÜV). For this acceptance procedure, the following software must be installed on your programmer:

- STEP 5 basic package
- COM 95F software package (parameterization software and acceptance tool)
- KOMDOK software package, if you have an earlier version of the STEP 5 basic package than V 6.0.

The Major Characteristics of the S5-95F and S5-115F at a Glance

Table 1-3. The Major Characteristics of the S5-95F and S5-115F at a Glance

Characteristics	S5-95F	S5-115F
Hardware Subunit dimensions (W×H×D) in mm	145×135×146	482×302×210
Supply voltage	24 V DC	24 V DC
Digital inputs - Failsafe onboard DIs - Failsafe external DIs - Non-failsafe external DIs	 16 max. 128 max. 320	 - max. 1024 max. 1024
Digital outputs - Failsafe onboard DQs - Failsafe external DQs - Non-failsafe onboard DQs - Non-failsafe external DQs	 8 max. 64 2×4 max. 320	 - max. 1008 - max. 1008
Analog inputs - Failsafe external AIs - Non-failsafe external AIs	 max. 16 max. 32	 max. 64 max. 64
Analog outputs - Non-failsafe external AQs	 max. 16	 max. 64
Failsafe, onboard counter inputs	2	-
Interrupt inputs - Failsafe onboard interrupt DIs (OB2) - Failsafe onboard interrupt DIs (OB3) - Failsafe external interrupt DIs (OB2)	 max. 4 max. 20 -	 - - 8
Interfaces	PG/OP/TD/SINEC L1	PG/OP/SINEC L1
Connection to SINEC L1 and PROFIBUS	yes	yes
Failsafe point-to-point link	no	yes
Software User memory (1 statement corresponds to 2 bytes)	16 KB, 8 KB of which typ. for program	37 KB
Operating system runtime - Without external I/Os - With external I/Os	25 ms 25 ms+2 ms per bus module	 approx. 80 ms
Run time of operating system	approx. 50 ms	approx. 80 ms
Execution time for 1024 binary statements	2 ms	2 ms
Number of flags - Retentive	2048 512	2032 -
Number of counters - Retentive	128 8	128 -
Number of timers	128	128
Program organization - Structured programming - Process interrupt handling - Time-controlled program processing - PID controller (non-failsafe)	 yes yes yes yes	 yes yes yes yes

2 Design, Functions and Operation

2.1	Basic System Design - without External I/Os	2 - 1
2.2	Basic System Design - with External I/Os	2 - 2
2.3	Internal Functions	2 - 4
2.3.1	Integral Real-Time Clock	2 - 4
2.3.2	Diagnostic Byte	2 - 4
2.3.3	Communication Byte	2 - 4
2.4	The Programmable Controllers' Principle of Operation	2 - 5
2.4.1	Functional Units	2 - 5
2.4.2	Mode of Operation of the External I/O Bus	2 - 8
2.5	Operating Instructions	2 - 11
2.5.1	Displays and Controls	2 - 11
2.5.2	Operating Modes	2 - 13
2.5.3	Operating Statuses of the S5-95F Following Power-Up	2 - 17
2.5.4	Performing an Overall Reset on the S5-95F	2 - 18
2.5.5	Function of the Backup Battery	2 - 19
2.5.6	Memory Submodules	2 - 20

Figures

2-1 S5-95F LEDs, Controls and Interfaces . . . . . 2 - 1
2-2 Basic Unit with External I/Os . . . . . 2 - 2
2-3 Functional Units of a Basic Programmable Controller . . . . . 2 - 5
2-4 Example of an Arithmetic Logic Unit's Mode of Operation . . . . . 2 - 7
2-5 Accumulator Design . . . . . 2 - 7
2-6 Structure of the External I/O Bus . . . . . 2 - 8
2-7 Data Cycle . . . . . 2 - 9
2-8 Displays and Controls I . . . . . 2 - 11
2-9 Displays and Controls II . . . . . 2 - 12

Tables

2-1 Retentive and Non-Retentive Operands . . . . . 2 - 7
2-2 Number of Bits per Module in the Shift Register . . . . . 2 - 10
2-3 Overview of LEDs . . . . . 2 - 12
2-4 Switching to Test Mode . . . . . 2 - 14
2-5 Switching to Quasi-Safety Mode . . . . . 2 - 15
2-6 Switching to Safety Mode . . . . . 2 - 16
2-7 Operating Statuses of the S5-95F Following Power-Up . . . . . 2 - 17
2-8 Overview of EPROM Submodules . . . . . 2 - 20

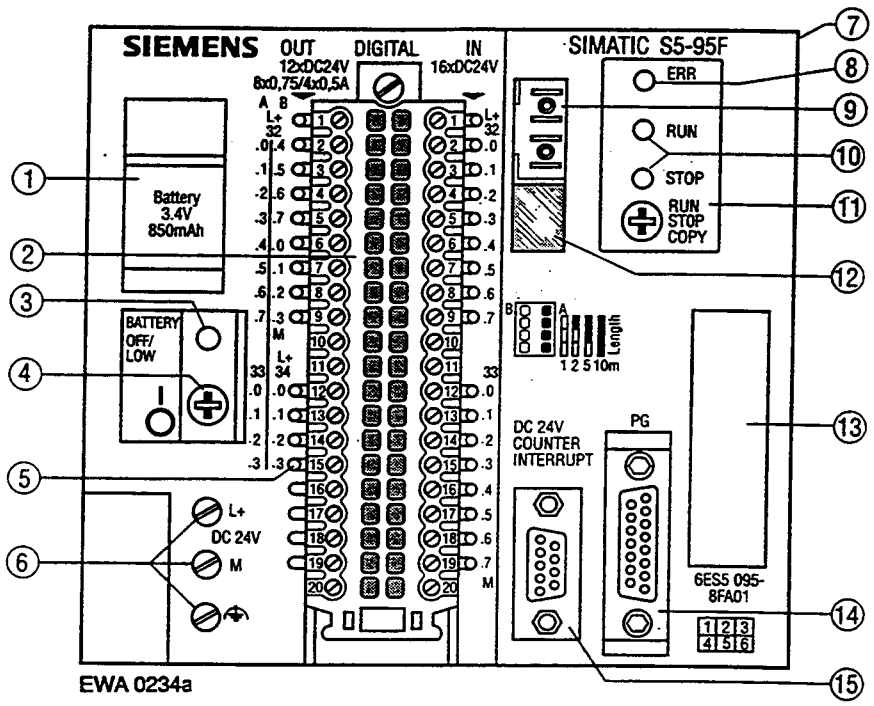
2 Design, Functions and Operation

This chapter contains information on the design and principle of operation of the S5-95F.

2.1 Basic System Design - without External I/Os

The following section discusses the basic unit without expansions. The basic unit has inputs and outputs available on board.

S5-95F Basic Unit



(1) Battery compartment
(2) Front connectors for digital inputs (I 32.0 to I 33.7) and for digital outputs (Q 32.0 to Q 32.7; Q 33.0 to Q 33.3 or Q 34.0 to Q 34.3)
(3) Battery failure display
(4) ON/OFF switch
(5) LED display for digital inputs and outputs
(6) Power supply terminals
(7) Connector for S5-100U bus units (not visible)
(8) Yellow error LEDs
(9) Connector for fiber optic cable
(10) Displays for operating mode: green LED → RUN; red LED → STOP
(11) Operating mode selector switch
(12) DIP switch for subunit identification
(13) Receptable for memory submodule
(14) Serial interface for programmer (PG), text display (TD), operator panel (OP) or SINEC L1 LAN
(15) Interface for interrupt inputs (I 59.0 to I 59.3) and for counter inputs (IW 36, IW 38)

Figure 2-1. S5-95F LEDs, Controls and Interfaces

2.2 Basic System Design - with External I/Os

You can expand the basic system by using S5-100U modules (external I/Os). These external I/Os consist of functional units that you can combine according to the task you want to perform.

Example of a Subunit with External I/Os

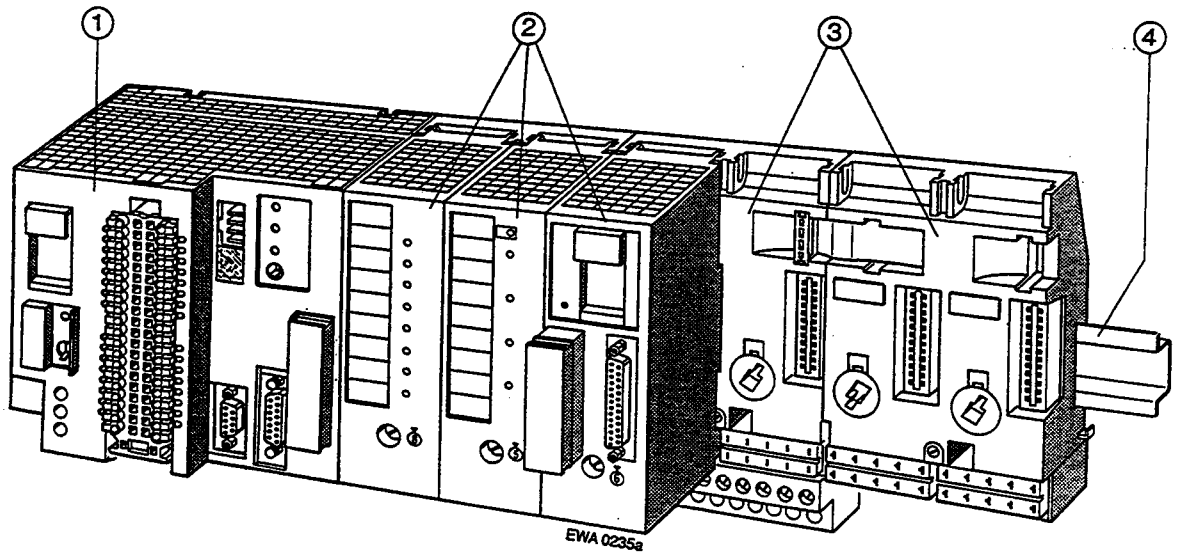


Figure 2-2. Basic Unit with External I/Os

① **Basic Unit**
For an S5-95F you always require two basic units. These two basic units are the core of the programmable controller and are connected via a fiber optic cable.

② **External I/Os**
If you require more than the available onboard I/Os or if you need special functions, you can expand your basic system with additional external I/Os.
The S5-95F can be expanded via bus units with S5-100U standard and failsafe modules (see Appendix A). A maximum of 8 bus units can be connected to each basic unit.

③ **Bus Units with Terminal Blocks** (crimp snap-in or SIGUT)
Bus units are used to connect the S5-100U modules to the basic unit. One bus unit can accommodate two I/O modules.

④ **Standard Mounting Rail**
The power supply module, the basic unit and the required bus units are mounted on the standard mounting rail.

Power Supply Module

The power supply module is required for the basic unit to use the 115 V/230 V AC supply voltages to generate the 24 V DC operating voltages.

External I/O Modules

The following external I/O modules can be used:

- Failsafe digital input and output modules
- Analog input modules for failsafe analog value processing
- Non-failsafe digital input and output modules from the S5-100U range
- Non-failsafe analog input and output modules from the S5-100U range
  You can use these modules to create and record variables (currents and voltages).
- Non-failsafe CP 521 SI communications modules from the S5-100U range
  The module enables
  - the output of event texts with date and time to a printer
  - connection of operator panels and text displays
  - and a link to other programmable controllers.
- Non-failsafe S5-100U function modules
  These modules make it possible to solve specific tasks, such as high-speed counting, positioning, closed-loop control, and many more.

2.3 Internal Functions

2.3.1 Integral Real-Time Clock

The integral real-time clock enables you to manipulate and control non-failsafe process sequences on a time-dependent basis, offering the following possiblities:

- Clock-time and calendar function
  This function allows you, for example, to determine exactly the point in time at which an error caused the S5-95F to go into the STOP mode.
- Interrupt or time interrupt function
  This function allows you, for example, to monitor the duration of a process.
- Operating hours counter
  This function allows you, for example, to monitor inspection intervals.

2.3.2 Diagnostic Byte

The diagnostic bytes (IB 35, 60 to 63) give you an additional possiblity to control the process sequence.

The diagnostic bytes display:

- Whether or not a counter has reached its comparison value
- Whether or not an interrupt was triggered
- Whether or not the power supply for the onboard I/Os has failed
- Whether or not battery backup is available.

2.3.3 Communication Byte

The communication bytes (IB 56 to 59) offer you the possibility to control the sequence of interrupt handling. You can evaluate the states of the interrupt bytes in the communication bytes.

2.4 The Programmable Controllers' Principle of Operation

This section describes how the controller processes your program.

2.4.1 Functional Units

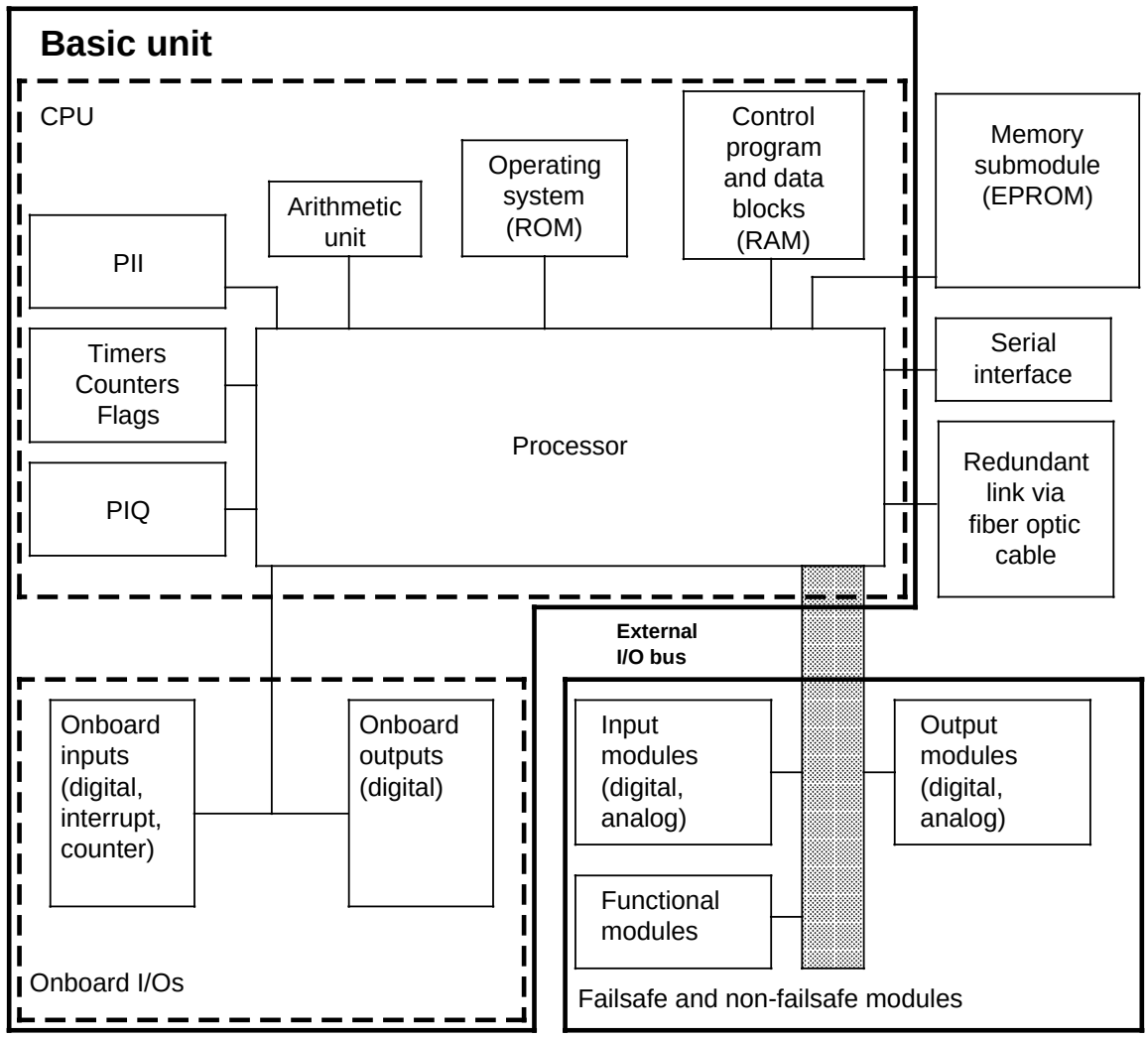


Figure 2-3. Functional Units of a Basic Programmable Controller

Program Memory (EPROM/RAM)

In order to safely store the control program outside of the PLC, you must store it on an EPROM or EEPROM memory submodule.
Programs that are available on a memory submodule (EPROM) can be copied to the internal program memory. This internal program memory is a reserved area of the CPU's internal RAM memory.

The internal RAM memory has the following characteristics:

- The memory contents can be changed quickly.
- Memory contents are lost when there is a supply voltage failure and there is no battery backup.

Operating System (ROM)

The operating system contains system programs that determine how the user program is executed, how inputs and outputs are managed, how the memory is divided, and how data is managed.
The operating system is fixed and cannot be changed.

Process Images (PII, PIQ)

Signal states of onboard inputs, onboard outputs, and output modules are stored in the CPU in "process images". Process images are reserved areas in the CPU RAM. During correct operation, the operating system ensures that the process images of both basic units are identical.

Input and output modules have the following separate images:

- Process image input table (PII)
- Process image output table (PIQ)

Serial Interface

Port for programmer, operator panels and monitors. Both basic units can be connected via this port either directly as slave to the SINEC L1 bus or to the PROFIBUS via a CP 541 communications processor.

Timers, Counters, Flags

The CPU has timers, counters and flags available internally that the control program can use.

The program can set, delete, start and stop the timers and counters. The time and count values are stored in the reserved areas of the RAM memory.

There is another area in the RAM memory where information such as intermediate results can be stored as flags. You can address the flags by bits, bytes, or words.

If battery backup is available, then some of the flags and counters remain in the internal RAM memory even if the supply voltage fails or the controller is switched off. These flags and counters are retentive.

Table 2-1 gives information about the number and retentive characteristics (the internal memory contents are retained/are not retained) of these timers, counters, and flags.

Table 2-1. Retentive and Non-Retentive Operands

Operand	Flags, Counters, Timers, Data Blocks and System Data	
	Retentive	Non-retentive
Flags	0.0 to 63.7	64.0 to 255.7
Counters	0 to 7	8 to 127
Timers	———	0 to 127
Data Blocks	1 to 255*	———
System Data	———	0 to 255

* Individual data words of DB252 to 254 are assigned default values by the operating system

Arithmetic Unit

The arithmetic unit consists of two accumulators, ACCU 1 and 2. The accumulators can process byte and word operations.

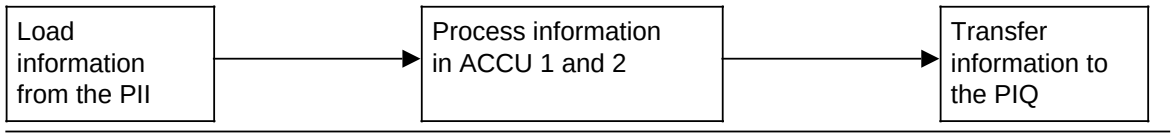


Figure 2-4. Example of an Arithmetic Logic Unit's Mode of Operation

Accumulator Design



Figure 2-5. Accumulator Design

Processor

According to the control program, the processor calls statements in the program memory in sequence and executes them. It processes the information from the PII and takes into consideration the values of internal timers and counters as well as the signal states of internal flags.

External I/O Bus

The I/O bus is the electrical connection for all signals that are exchanged between the basic unit external I/O and the S5-100U modules.

2.4.2 Mode of Operation of the External I/O Bus

The programmable controllers have a serial bus for the transfer of data between the CPU and the S5-100U modules. This serial bus has the following characteristics.

- The modular design permits optimal adaptation to the particular control task.
- No addresses have to be set on the I/O modules.
- A terminating resistor connector is not required.
- Direct access to individual modules is not possible.

The data is moved via a number of shift registers (see Figure 2-6).

Four data bits and one check bit for bus monitoring are assigned to each slot in the bus unit. All modules requiring more than four data bits have their own shift register and therefore do not have to use the shift register of the particular slot.

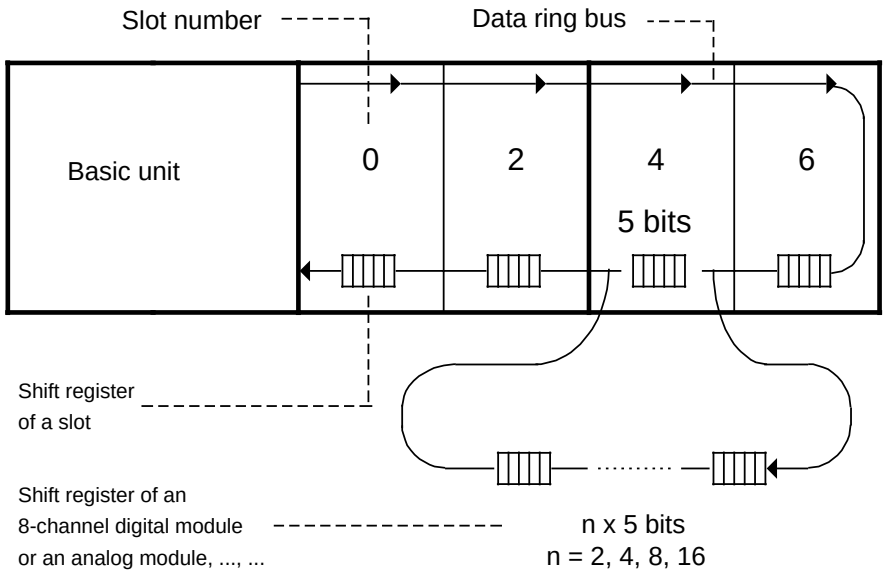


Figure 2-6. Structure of the External I/O Bus

Data Cycle

Prior to a program scan, the external I/O bus transfers the current information of the input modules to the process image input table (PII). At the same time, the information contained in the process image output table (PIQ) is transferred to the output modules.

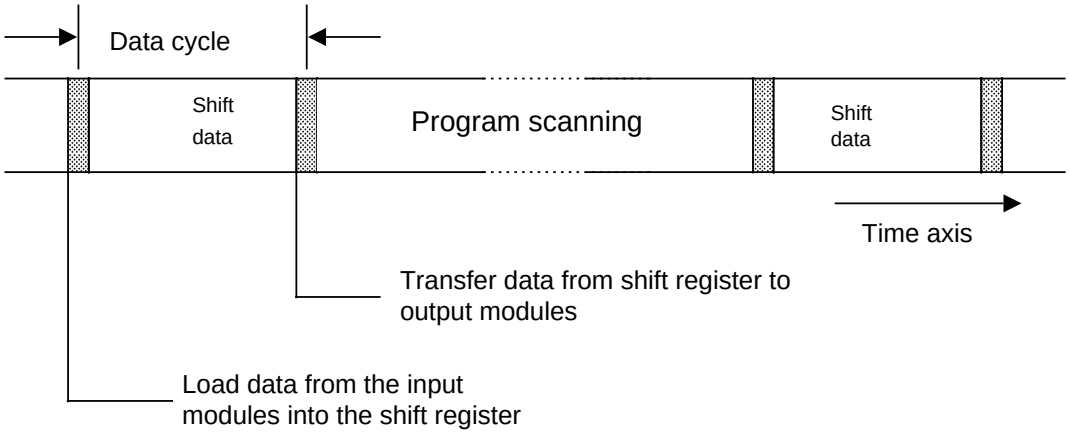


Figure 2-7. Data Cycle

Interrupt Data Cycle

There is an interrupt input data cycle prior to each time-controlled program scan.
Before a time-controlled program scan (OB 13), current information of the input modules is read into the interrupt PII.

Following a time-controlled program scan (OB 13), there is not an interrupt output data cycle until data has been moved into the interrupt PIQ via a transfer operation (see section 8.1.3).
Information is output from the interrupt PIQ to the output modules during an interrupt output data cycle. The PIQ is updated.

Length of the Shift Register

The total length of the shift register is obtained from the sum of the data bits of all plugged-in modules and of the empty slots. The check bit is not counted.

You must know the length of the shift register to be able to determine the data cycle time. Data cycle time is 25 µs x number of data bits.

Table 2-2. Number of Bits per Module in the Shift Register

Plugged-in Module	Number of Data Bits
Failsafe modules	
Digital input module	8
Digital output module	8
Non-failsafe modules (e. g.)	
Digital input/output modules, 8-channel	8
Digital input and output module,16DI/16DQ	16
Analog modules for each switched channel	16
CP 521 SI communication module	64
Vacant slot	4

2.5 Operating Instructions

The following sections provide important information for successful system startup.

They describe:

- Displays and controls
- Operating modes and statuses
- Overall reset
- Permissible memory submodules

2.5.1 Displays and Controls

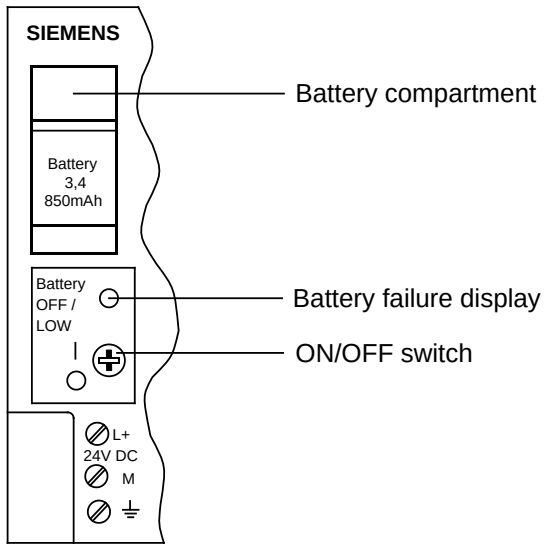


Figure 2-8. Displays and Controls I

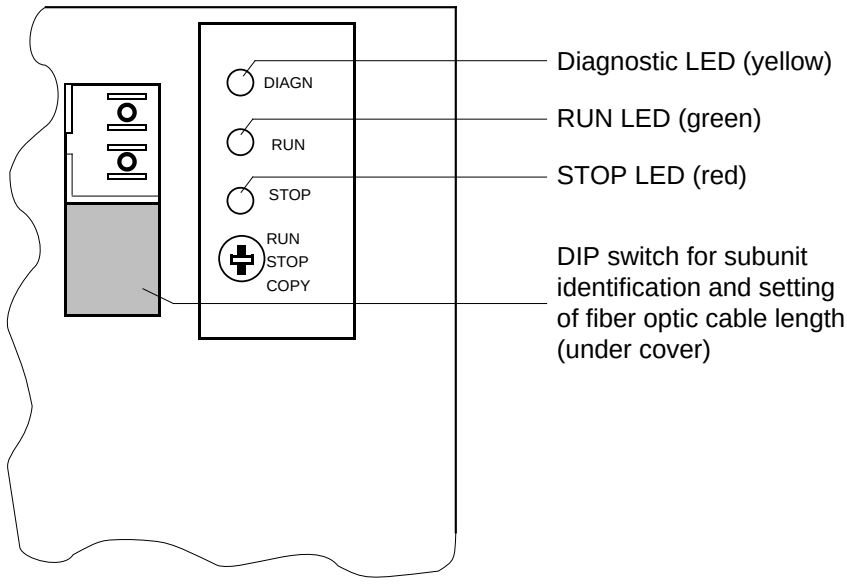


Figure 2-9. Displays and Controls II

Overview of LEDs

Table 2-3. Overview of LEDs

RUN LED	STOP LED	Error LED	Meaning
1. Subunit: on 2. Subunit: on	1. Subunit: off 2. Subunit: off	1. Subunit: off 2. Subunit: off	System in RUN mode
1. Subunit: on 2. Subunit: on	1. Subunit: off 2. Subunit: off	1. Subunit: on 2. Subunit: on	System in RUN mode with error
1. Subunit: flashing 2. Subunit: off	1. Subunit: on 2. Subunit: on	1. Subunit: off 2. Subunit: off	First unit is ready and waiting for second unit to be switched to RUN
1. Subunit: off 2. Subunit: off	1. Subunit: flickering 2. Subunit: flickering	1. Subunit: off 2. Subunit: off	Subunits are synchronized
1. Subunit: on 2. Subunit: on	1. Subunit: On 2. Subunit: on	1. Subunit: off 2. Subunit: off	System is in the startup self-test
1. Subunit: off 2. Subunit: off	1. Subunit: on 2. Subunit: on	1. Subunit: on 2. Subunit: on	System with error in STOP mode
1. Subunit: off 2. Subunit: off	1. Subunit: on 2. Subunit: on	1. Subunit: off 2. Subunit: off	System in STOP mode
1. Subunit: off 2. Subunit: off	1. Subunit: on 2. Subunit: on	1. Subunit: flashing 2. Subunit: flashing	System hard STOP Overall reset required

2.5.2 Operating Modes

The S5-95F distinguishes between three operating modes:

- Test mode
- Quasi-safety mode
- Safety mode

> **Safety Note**
>
> Whenever the process controller executes safety functions, the S5-95F must operate in the "safety mode". The "test mode" and "quasi-safety mode" are for the exclusive purpose of testing the user program.

Self-Test during STARTUP

High-grade self-tests for most of the components are integrated in the operating system of the S5-95F; for all other components, at least one standard self-test is provided. The duration of the total self-test routine is approx. 60 s. It is executed as a whole only during STARTUP.

Self-Test during Cyclic Operation

For reasons of time, the S5-95F executes only a small part of the self-test routine at a time during cyclic operation. That is why the self-test routine is divided into several test slices. At the end of cyclic program execution, the S5-95F automatically executes one or several of these test slices. The whole self-test routine is thus executed once within one hour.

Test Mode

If no ***EPROM submodule*** is plugged in, the S5-95F is automatically in the test mode. Via the programmer you transfer the control program directly into the internal RAM of the S5-95F.

The test mode is used only to test the control program. You can use all programmer functions without any restrictions.

Self-test routine in test mode
In the STARTUP mode, the S55-95F skips the execution of the self-test. In the RUN mode, it executes the self-test in test slices. The operating system of the S5-95F ensures that all test slices are executed once within one hour.

Switching the S5-95F to Test Mode

Prerequisites:

- The ON/OFF switches of both basic units are in the "0" position
- The RUN/STOP switches of both basic units are in the STOP position
- ***No EPROM submodules*** are plugged into the two basic units

Table 2-4. Switching to Test Mode

Step	Procedure	S5-95F Reaction
1	Switch on the power supply of both subunits	The subunits are synchronized.
	Set ON/OFF switch to "I"	Both STOP LEDs flicker for approx. 20 s; afterwards, both STOP LEDs show steady light.
2	Switch the system to RUN	
	Set the RUN/STOP switch of subunit A to RUN	Subunit A: STOP LED on, RUN LED flashing Subunit B: STOP LED on
	Set the RUN/STOP switch of subunit B to RUN (within 30 min.)	STOP and RUN LEDs in both subunits light up briefly; afterwards both RUN LEDs show steady light. System is in test mode.

Quasi-Safety Mode

The S5-95F is in the quasi-safety mode, if no ***EPROM submodule*** is plugged in ***and*** when you observe a ***special sequence*** when switching on the system. Via the programmer you transfer the control program directly into the internal RAM of the S5-95F.

In the quasi-safety mode, the S5-95F operates in the same way as in the safety mode. Since, however, the control program is not protected against corruption, the quasi-safety mode may only be used for testing of the control program.

Self-test routine in quasi-safety mode
Immediately after switching on the quasi-safety mode, the S5-95F executes the complete self-test routine and thus checks all components. Execution of the total self-test routine takes approx. 60 s. In the RUN mode, the S5-95F executes the self-test in test slices. The operating system of the S5-95F ensures that all test slices are executed once within one hour.

Switching the S5-95F to Quasi-Safety Mode

Prerequisites:
- The system has executed the same control program at least once in the test mode
- The ON/OFF switches of both basic units are in the "0" position
- The RUN/STOP switches of both basic units are in the STOP position
- ***No EPROM submodules*** are plugged into the two basic units

Table 2-5. Switching to Quasi-Safety Mode

Step	Procedure	S5-95F Reaction
1	Switch on the power supply of both subunits	The subunits are synchronized.
	Set ON/OFF switch to "I"	Both STOP LEDs flicker for approx. 20 s; afterwards, both STOP LEDs show steady light.
2	Switch the system to RUN	
	Briefly press the RUN/STOP switch of subunit A and B simultaneously in the COPY position	System switches to quasi-safety operation.
	Set the RUN/STOP switch of subunit A to RUN	Subunit A: STOP LED on, RUN LED flashing Subunit B: STOP LED on
	Set the RUN/STOP switch of subunit B to RUN (within 30 min.)	System executes complete self-test, STOP and RUN LEDs in both subunits are on for approx. 60 s, afterwards both RUN LEDs show steady light.
		System is in quasi-safety mode

If you wish to change from quasi-safety mode to test mode, you must perform an overall reset on the S5-95F.

Safety Mode

The S5-95F is automatically in the safety mode, if the control program including DB1 is stored on an EPROM submodule and if both EPROM submodules are plugged in. Programmer operation is restricted in the safety mode.

Self-test routine in safety mode
Immediately after switching on the safety mode, the S5-95F executes the complete self-test routine and thus checks all components. Execution of the total self-test routine takes approx. 60 s. In the RUN mode, the S5-95F executes the self-test in test slices. The operating system of the S5-95F ensures that all test slices are executed once within one hour.

Switching the S5-95F to Safety Mode

Prerequisites:

- The ON/OFF switches of both basic units are in the "0" position
- The RUN/STOP switches of both basic units are in the STOP position
- EPROM submodules are plugged into both basic units

Table 2-6. Switching to Safety Mode

Step	Procedure	S5-95F Reaction
1	Switch on the power supply of both subunits	The subunits are synchronized.
	Set ON/OFF switch to "I"	Both STOP LEDs flicker for approx. 20 s; afterwards, both STOP LEDs show steady light.
2	Switch the system to RUN	
	Set the RUN/STOP switch of subunit A to RUN	Subunit A: STOP LED on, RUN LED flashing Subunit B: STOP LED on
	Set the RUN/STOP switch of subunit B to RUN (within 30 min.)	System executes complete self-test, STOP and RUN LEDs in both subunits are on for approx. 60 s, afterwards both RUN LEDs show steady light.
		System is in safety mode

2.5.3 Operating Statuses of the S5-95F Following Power-Up

Table 2-7. Operating Statuses of the S5-95F Following Power-Up

Operating Status	Prerequisite	Characteristics of the Operating Status	Optical Display
System initialization	ON/OFF switches of both subunits in "1" position	Subunits are synchronized • All outputs are disabled • In the safety mode, the S5-95F loads the control program from the memory submodule • In the test mode, the control program is not loaded automatically	STOP LEDs flickering
STOP	System initialization executed successfully Position of RUN/STOP switches irrelevant	• Self-tests are executed, but no I/O test • All outputs are disabled • User program is not yet processed	STOP LEDs show steady light
STARTUP	RUN/STOP switches of both subunits in "RUN" position	• Execution of the complete self-test in the safety mode • Process I/O images, timers and non-retentive flags and counters are set to zero • DB1 parameters are accepted • OB21/22 are processed • External I/Os and onboard I/Os are not yet addressable • Scan monitor is switched off • Time-controlled and interrupt-controlled processing are not possible	STOP and RUN-LEDs show steady light
RUN	STARTUP executed successfully Both RUN/STOP switches in "RUN" position	• Execution of the complete additional self-test routine once within one hour • Process I/O images, timers, counters and flags are compared cyclically • Scan monitor is active • OB1 is processed • Input modules are read • Output modules are addressed • Time-controlled and interrupt-driven processing not possible	RUN LEDs show steady light

Please note in case of a slow, continuous voltage drop:

> **Note:**
>
> If the 24 V DC supply drops slowly (changes from 17 V to 14 V within 1 s), the S5-95F detects an I/O fault and does not restart after power recovery. It has to be started up via the RUN/STOP switch.

2.5.4 Performing an Overall Reset on the S5-95F

You should perform an overall reset

- before you load a new control program into the S5-95F
- if the S5-95F requests an overall reset because of an error, e.g. during system initialization. You are requested to perform an overall reset when the yellow error LED flashes regularly.

The S5-95F offers you three different ways of performing an overall reset:

- Overall reset via programmer
- Manual reset
- Automatic reset

Overall Reset via Programmer

You can select the overall reset function from the programmer's menu line (refer to the programmer manual).
Please note that an overall reset via the programmer does not delete all data in both subunits. The following data is retained:

- Programmer bus number
- Clock data of the integral real-time clock
- Internal data for system protection (e.g. safety procedure after power failure)

Manual Reset

With a manual reset, the complete data in both subunits is deleted.

To perform a manual reset, you should observe the following

Set both RUN/STOP switches to the STOP position
Set both ON/OFF switches to "0"
Remove batteries from both basic units
Set both ON/OFF switches to "I"
Insert the batteries in both basic units
Switch system to RUN mode

Safety Note

For reasons of electromagnetic compatibility, the ESD protective measures (see Appendix D) must be observed when handling batteries.

Automatic Reset

The S5-95F executes an automatic reset when the memory submodules are replaced by memory submodules of different type or size.

When the programmable controller executes an automatic reset, the following data is retained:

- Programmer bus number
- Clock data of the integral real-time clock
- Internal system backup data (e.g. Safety procedure after power failure)
- System event DB
- System ID number

2.5.5 Function of the Backup Battery

A backup battery is absolutely necessary for S5-95F operation. If the power fails or if the subunits are switched off, the contents of the internal RAM are retained.

The S5-95F monitors the status of the battery. If the battery charge drops below a permissible value during operation, the operating system sets a bit in the system event DB; OB34 is then called prior to each scan cycle. If the battery is not replaced within 72 hours after this message, the S5-95F goes to STOP.

If one battery is missing or if the battery charge drops below a permissible value, you cannot switch the S5-95F to the RUN mode after a power OFF/power ON sequence.

Replacing the Backup Battery

Do not wait until the S5-95F indicates that it is time to change the battery, simply replace it every year as a preventive measure.

Safety Note

Battery replacement in the RUN mode is not permissible for systems under the authority of the German Statutory Industrial Accident Insurance Institution (BG) or the BG Institute for Industrial Safety (BIA). In all other systems you must observe the ESD guidelines when replacing a battery in the RUN mode.

Disposal of Backup Batteries



Warning

Do not charge lithium batteries. They could explode. Dispose of used batteries properly.

2.5.6 Memory Submodules

If you want to operate the S5-95F in the safety mode, the control program must be stored on an EPROM submodule. The following table lists the permissible memory submodules.

Table 2-8. Overview of EPROM Submodules

Type of Submodule	Order Number	Capacity	Programming No.
EPROM	6ES5 375-0LA15	8 KB	11
EPROM	6ES5 375-0LA21	16 KB	12
EPROM	6ES5 375-0LA41	32 KB	17
EPROM	6ES5 375-1LA15	8 KB	411
EPROM	6ES5 375-1LA21	16 KB	412
EPROM	6ES5 375-1LA41	32 KB	417

Note:

For operation with memory submodules you always require two identical memory submodules. Make sure that both submodules have the same order number.

3 Guidelines for Planning and Installation of the Product

3.1	Guidelines on the Safe Integration of the Product into its Environment	3 - 1
3.2	Installation of Programmable Controllers in Accordance with Principles of EMC	3 - 2
3.2.1	Overview of Possible Interference	3 - 2
3.2.2	Most Important Basic Rules for Ensuring EMC	3 - 4
3.3	Installation of Programmable Controllers for EMC	3 - 5
3.3.1	Basic Rules for Assembling and Grounding the Inactive Metal Parts	3 - 5
3.3.2	Example of Rack Mounting for EMC	3 - 6
3.4	Wiring of Programmable Controllers for EMC	3 - 7
3.4.1	Routing of Cables	3 - 7
3.4.2	Equipotential Bonding	3 - 9
3.4.3	Shielding of Cables and Lines	3 - 10
3.4.4	Special Measures for Interference-Free Operation	3 - 11
3.4.5	Filters for 24 V DC Power Supply Units	3 - 13
3.4.6	Checklist for the Electromagnetically Compatible Installation of Control Systems	3 - 14

Figures

3-1 Electromagnetic Interference with Programmable Controllers . . . . . 3 - 2
3-2 Rack Mounting of an S5-95F Two-Tier Configuration . . . . . 3 - 6
3-3 Routing of Equipotential Bonding Conductor and Signal Line . . . . . 3 - 9
3-4 Examples of Securing Shielded Lines with Cable Clamps . . . . . 3 - 11
3-5 Quenching Circuits for Inductances . . . . . 3 - 11
3-6 Measures for Interference Suppression of Fluorescent Lamps in a Cabinet . . . . . 3 - 12
3-7 Filter for Protection to IEC 801-5, Severity III . . . . . 3 - 13

Tables

3-1 Interference Sources and Their Causes . . . . . 3 - 3
3-2 Rules for Laying of Line Combinations . . . . . 3 - 7
3-3 Components for Network Filters . . . . . 3 - 13
3-3 Checklist for Electromagnetically Compatible Installation . . . . . 3 - 14

3 Guidelines for Planning and Installation of the Product

The following sections contain instructions on the planning and installation of systems or plants equipped with programmable controllers.

3.1 Guidelines on the Safe Integration of the Product into its Environment

Since the product generally forms part of a larger system or plant, these guidelines are intended to help integrate the product into its environment without it constituting a source of danger.



Warning

- Make sure that the conditions of inspection of the S5-95F and the instructions for safety-related use of the S5-95F contained in this manual are adhered to.
- Strictly follow the safety and accident prevention rules that apply in each particular case.
- In the case of equipment with a permanent power connection which is not provided with an isolating switch and/or fuses which disconnect all poles, a suitable isolating switch or fuses must be provided in the building wiring system (distribution board). Furthermore, the equipment must be connected to a protective ground (PE) conductor.
- Before switching on the equipment, make sure that the voltage range setting on the equipment corresponds to the local power system voltage.
- In the case of equipment operating on 24 V, make sure that proper electrical isolation is provided between the mains supply and the 24V supply. Use only power supply units complying with DIN VDE 0551 or manufactured in accordance with DIN VDE 0551 (or EN 60742) and DIN VDE 0160. Also strictly follow the conditions of electromagnetic compatibility (EMC) (see Chapter 3.2 ff).
- Fluctuations or deviations of the power supply voltage from the rated value should not exceed the tolerances specified in the technical specifications. Otherwise, functional failures or dangerous conditions can occur in the electronic modules/equipment.
- Suitable measures must be taken to make sure that programs that are interrupted by a voltage dip or power supply failure resume proper operation when the power supply is restored. Care must be taken to ensure that dangerous operation conditions do not occur even momentarily. If necessary, the equipment must be forced into the "emergency off" state.
- Emergency tripping to devices in accordance with EN 60204/IEC 204 (VDE 0113) must be effective in all operating modes of automation equipment. Resetting the emergency off device must not result in any uncontrolled or undefined restart of the equipment.
- Install the power supply and signal cables in such a manner as to prevent inductive and capacitive interference voltages from affecting the automation functions.
- Automation equipment and its operating elements must be installed in such a manner as to prevent unintentional operation.
- Automation equipment can assume an undefined state in the case of wire break in the signal lines. To prevent this, suitable hardware and software measures must be taken when interfacing the inputs and outputs of the automation equipment.
- For applications within the framework of the Professional/Trade Associations guidelines or for applications to which EN 60204 applies, the compartment in which the S5-95F is installed must have IP 54 protection.

3.2 Installation of Programmable Controllers in Accordance with Principles of EMC

Interference-Free Installation of the S5-95F

What Does EMC Mean?
Electromagnetic compatibility (EMC) is understood to mean the capability of electrical equipment to operate correctly in a defined electromagnetic environment, without being affected by the environment and without affecting the environment to an unacceptable degree.

All SIMATIC S5 products have been developed for applications in harsh industrial environments and meet high requirements for EMC. Before installing the control system, however, you should still carry out EMC planning and involve possible interference sources in the assessment.

Described in the following section are:

- The various paths over which interference can be picked up in the PLC
- Typical interference sources and their coupling mechanisms
- Basic rules for ensuring EMC.

3.2.1 Overview of Possible Interference

Electromagnetic interference can be picked up by the programmable controller over different paths (see Figure 3-1).

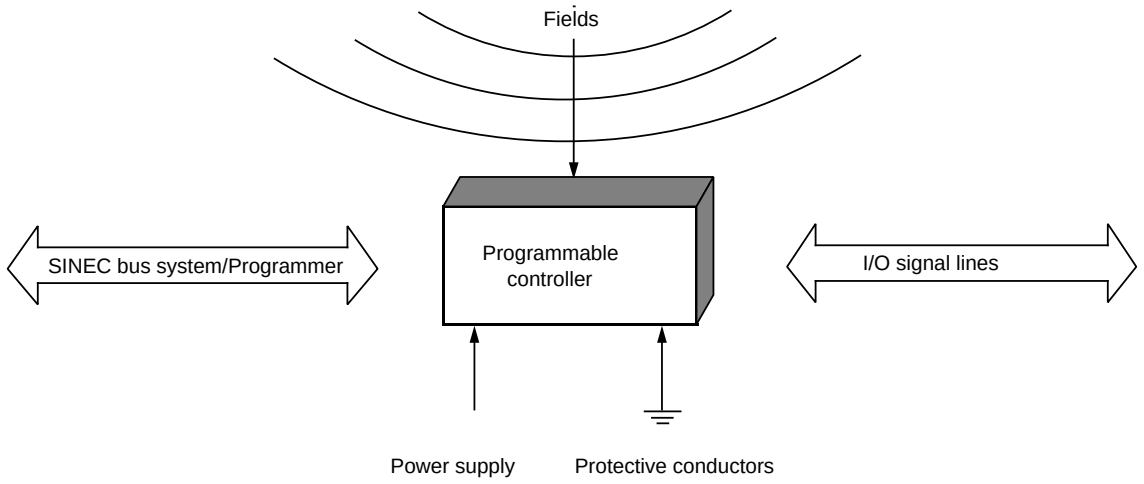


Figure 3-1. Electromagnetic Interference with Programmable Controllers

Depending on the propagation medium (conducted or non-conducted interference) and distance from the source, interference can be picked up by the programmable controller via different coupling mechanisms.

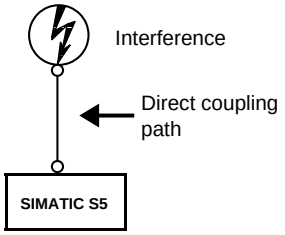
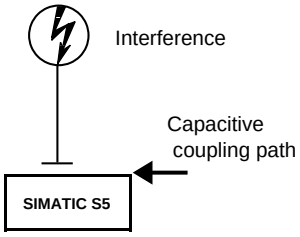
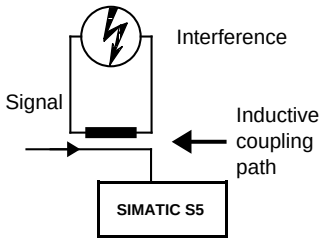
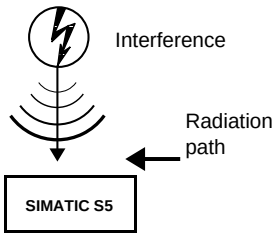
A distinction is made between the following:

- Direct coupling
- Capacitive coupling
- Inductive coupling
- Radiated interference

Shown in the following table are the four different coupling mechanisms, their causes and possible interference sources.

Coupling Mechanisms and Typical Interference Sources at a Glance

Table 3-1. Interference Sources and Their Causes

Coupling Mechanism	Cause	Typical Interference Source
• **Direct Coupling** 	Direct or metallic coupling always occurs when two circuits have a common conductor.	• Switched devices (supply affected by inverters and external power supply units) • Motors being started • Different potentials of component cases with a common power supply • Static discharges
• **Capacitive Coupling** 	Capacitive or electrical coupling occurs between conductors which are at different potentials. The degree of coupling is proportional to the voltage variation as a function of time.	• Interference pickup via parallel signal cables • Static discharge of the operator • Contactors
• **Inductive Coupling** 	Inductive or magnetic coupling occurs between two conductor loops through which current is flowing. Interference voltages are induced by the magnetic fluxes associated with the currents. The degree of coupling is proportional to the current variation as a function of time.	• Transformers, motors, electric welders • Parallel AC supply cables • Cables with switched currents • Signal cables with a high frequency • Unconnected coils
• **Radiated Interference** 	There is a radiation path when a conductor is subjected to an electromagnetic wave. Impinging of the wave results in induced currents and voltages.	• Local transmitters (e.g. two-way radios) • Spark gaps (spark plugs, collectors of electric motors, welders)

3.2.2 Most Important Basic Rules for Ensuring EMC

When installing the control system, please observe the following five basic rules.

When installing the programmable controllers, provide large-area good quality grounding of the inactive metal parts (see section 3.3.1)

- Make a large-area low-impedance interconnection of all inactive metal parts.
- For screw connections on painted and anodized metal parts, either use special contact washers or remove the insulating protective layers.
- If possible, do not use aluminium parts. Aluminium oxidizes easily and is therefore less suitable for grounding.
- Make a central connection between the chassis ground and the ground/protective ground conductor system.

Ensure proper routing of lines when wiring (see sections 3.4.1 and 3.4.2)

- Arrange the cabling in line groups.
  (AC power cable, power supply lines, signal lines, data lines).
- Always install AC power cables and signal or data lines in separate ducts or bunches.
- Route the signal and data lines as closely as possible to grounded surfaces such as cabinet elements, metal bars and cabinet panels.

Ensure that cable shields are properly secured (see section 3.4.3)

- Data lines must be shielded. The shield should be connected at both ends.
- Analog lines must be shielded. For the transfer of signals with low amplitudes, it may be advisable to connect the shield at only one end.
- Provide the line shields with a large-area connection to a shield/protective conductor bar immediately after the cabinet inlet, and secure the shields with cable clamps. Route the grounded shield as far as the module without interruption, but do not connect the shield there again.
- Ensure that the shield/protective ground bar has a low-impedance connection to the cabinet.
- Use metal or metallized connector cases for shielded data lines.

Employ special EMC measures for particular applications (see sections 3.4.4 and 3.4.5)

- Fit quenching elements not controlled by SIMATIC S5 modules to all inductances.
- Use incandescent bulbs for illuminating cabinets, and avoid fluorescent lamps.
- Filter the 24 V DC supply for onboard interrupt DIs/counter inputs, the DI 431-8FA11 digital input module, and the DQ 450-8FA11 digital output module.

Create a standard reference potential; ground all electrical apparatus if possible

- Use specific grounding measures. Grounding of the control system is a protective **and** functional measure.
- System parts and cabinets with central controllers and expansion units should be connected to the ground/protective conductor system in star configuration. This serves to avoid the creation of ground loops.
- In the case of potential differences between system parts and cabinets, install equipotential bonding conductors of sufficient rating.

3.3 Installation of Programmable Controllers for EMC

Measures for suppressing interference voltages are often applied only when the control system is already operational and proper reception of a useful signal is impaired. The reason for such interference is usually inadequate reference potentials caused by mistakes in equipment assembly.

Described in the following sections are:

- Basic rules for grounding the inactive metal parts
- Examples of cabinet assembly for EMC
- Example of rack and wall mounting for EMC

3.3.1 Basic Rules for Assembling and Grounding the Inactive Metal Parts

Ensure wide-area chassis grounding of the inactive metal parts when mounting the equipment.

Properly implemented chassis grounding creates a uniform reference potential for the control system, and reduces the effects of picked-up interference.

Chassis grounding is understood to mean the electrical connection of all inactive parts. The entirety of all interconnected inactive parts is the chassis ground.

Inactive parts are conductive parts which are electrically isolated from active parts by basic insulation, and can only develop a voltage in the event of a fault.

The chassis ground ***must not develop a dangerous touch voltage***, even in the event of a fault. The ground must therefore be connected to the protective ground conductor. To prevent ground loops, locally separated ground elements such as cabinets, structural and machine parts, must always be connected to the protective ground system ***in star configuration***.

Ensure the following when chassis grounding:

- Connect the inactive metal parts with the same degree of care as the active parts.
- Ensure low-impedance metal-to-metal connections, e.g. with large-area good quality contact.
- When you are incorporating painted or anodized metal parts in the grounding, these insulating protective layers must be penetrated. Use special contact washers or remove the insulating layer.
- Protect the connection points from corrosion, e.g. with grease.
- Movable grounded parts such as cabinet doors must be connected via flexible grounding strips. The grounding strips should be short and have a large surface because the surface is decisive in providing a path to ground for high-frequency interference.

3.3.2 Example of Rack Mounting for EMC

Picked-up interference should pass to large metal surfaces. You should therefore secure standard mounting rails, shield and protective conductor bars to metal structural elements. For wall mounting in particular, installation on reference potential surfaces made of sheet steel has proved advantageous.

When installing shielded cables, provide a shield bar for connecting the cable shields. The shield bar can also be the protective conductor bar.

Ensure the following for rack and wall mounting:

- Use only standard mounting rails to EN 50022 to mount the modules of your S5-95F. Only these mounting rails guarantee a tight fit of the modules and a satisfactory discharge of picked-up interference.
- Suitable contacting aids should be used on painted and anodized metal parts. Use special contact washers or remove the insulating protective layers.
- Provide large-area, low-impedance metal-to-metal connections when securing the shield/protective conductor bar.

Rack Mounting

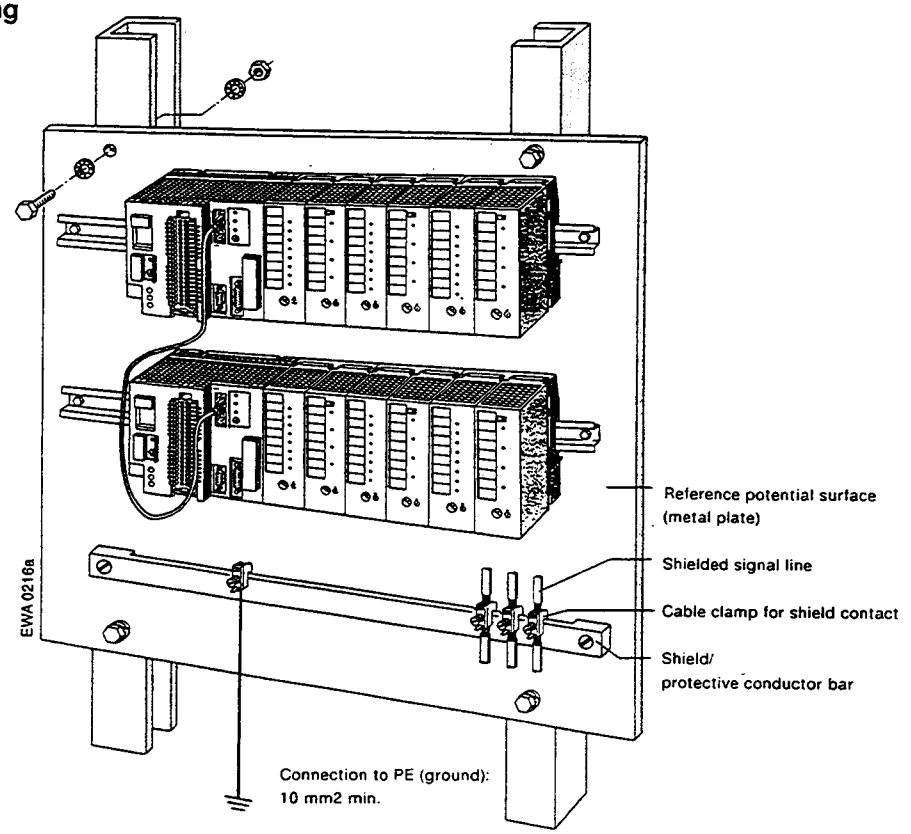


Figure 3-2. Rack Mounting of an S5-95F Two-Tier Configuration

3.4 Wiring of Programmable Controllers for EMC

The following section describes:

- Routing of cables within and outside cabinets
- Equipotential bonding between devices
- Single and double-ended connection of cable shields
- Checklist for electromagnetically compatible installation

3.4.1 Routing of Cables

This section covers the routing of bus, signal and supply lines. The purpose of cable routing is to suppress crosstalk between cables laid in parallel.

Routing of Cables Within and Outside Cabinets

For electromagnetically compatible routing of cables and lines, it is advisable to subdivide the lines for S5-95F and third-party systems into the following line groups and lay the groups separately.

Group A: Shielded bus and data lines (for programmer, OP, SINEC L1, PROFIBUS, printer, etc.)
Shielded analog lines
Unshielded lines for DC voltage 60 V
Unshielded lines for AC voltage 25 V
Coaxial cables for monitors

Group B: Unshielded lines for DC voltage > 60 V and 400 V
Unshielded lines for AC voltage > 25 V and 400 V

Group C: Unshielded lines for DC and AC voltage > 400 V

Group D: Cables for Industrial Ethernet

From the combination of individual groups in the following table, you can read off the conditions for laying the line groups.

Table 3-2. Rules for Laying of Line Combinations

	Group A	Group B	Group C	Group D
Group A				
Group B				
Group C				
Group D				

Legend for the table:

Lines can be laid in common bundles or cable ducts.
Lines must be laid in separate bundles or cable ducts (without minimum clearance).
Lines within cabinets must be laid in separate bundles or cable ducts; outside the cabinets but within buildings, they must be laid over separate cable routes with a clearance of at least 10 cm (3.93 in.).
Lines must be laid in separate bundles or cable ducts with a clearance of at least 50 cm (1.64 ft.).

Routing of Cables Outside Buildings

Outside buildings, lay the lines on metal cable trays if possible. Provide the joints between cable trays with an electrical connection and ground the cable trays.

When laying lines outside buildings, you must observe the valid lightning protection and grounding measures. The following applies in general:

Lightning Protection

Where cables and lines for SIMATIC S5 controllers are to be laid outside buildings, you must apply measures for internal and external lightning protection.

Outside the buildings, lay your lines either

- in metal conduits grounded at both ends,
  or
- in concreted cable ducts with continuously connected reinforcement.

Protect the signal lines from overvoltages by means of

- varistors
  or
- inert gas-filled surge diverters.

Fit these protective devices at the cable entry into the building.

> **Note**
>
> Lightning protection measures always require an individual assessment of the entire installation. For clarification, please consult your local Siemens office or a company specializing in lightning protection, such as Messrs. Dehn und Söhne in Neumarkt (Germany).

Equipotential Bonding

Ensure adequate equipotential bonding between the connected equipment (see section 3.4.2).

3.4.2 Equipotential Bonding

Between separate sections of an installation, potential differences can develop if

- programmable controllers and I/O devices are connected via non-floating links,
  or
- cable shields are connected at both ends and are grounded at different parts of the system.

Different AC supplies, for example, can cause potential differences. These differences must be reduced by installing equipotential bonding conductors to ensure functioning of the electronic components.

The following points must be observed for equipotential bonding:

- The lower the impedance of the equipotential bonding conductor, the greater is the effectiveness of equipotential bonding.
- Where shielded signal lines are laid between the relevant sections of the system and connected at both ends to the ground/protective conductor, the impedance of the additional equipotential bonding conductor must not exceed 10 % of the shield impedance.
- The cross-section of the equipotential bonding conductor must be rated for the maximum circulating current. The following cross-sections of copper have proved to be satisfactory in practice:
  - 16 mm^2 for equipotential bonding conductors of up to 200 m (656 ft.) in length
  - 25 mm^2 for equipotential bonding conductors of more than 200 m (656 ft.) in length.
- Use copper or zinc-plated steel for equipotential bonding conductors. They must be given a large-area connection to the ground/protective conductor and protect it from corrosion.
- The equipotential bonding conductor should be laid so that the smallest possible areas are enclosed between the equipotential bonding conductor and signal lines (see section 3.3).

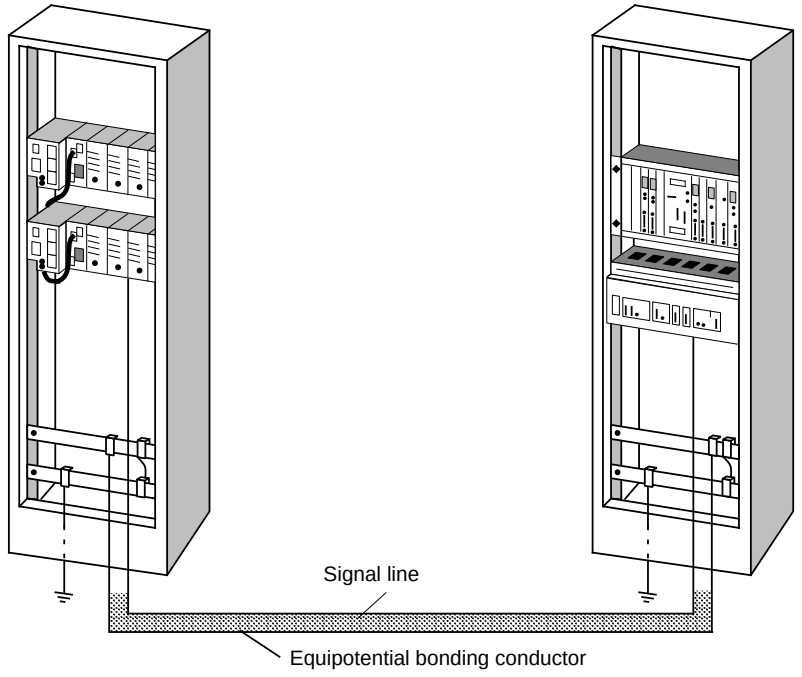


Figure 3-3. Routing of Equipotential Bonding Conductor and Signal Line

3.4.3 Shielding of Cables and Lines

Shielding is a method of attenuating magnetic, electrical or electromagnetic interference fields.

Interference currents on cable shields are passed to ground via the shield bar which is electrically connected to the housing. A low-impedance connection to the protective conductor is particularly important so that these interference currents themselves do not become an interference source.

Where possible, only use lines with a braided shield. The coverage density of the shield should be more than 80 %. Avoid lines with a foil shield because the foil can be very easily damaged by tensile strain and compression during fitting; this results in reduced effectiveness of the shield.

As a rule, line shields should always be connected at both ends. This is the only way to achieve a good degree of interference suppression in the higher frequency region.

Only in exceptional cases should the shield be connected at one end only, as this only achieves attenuation of the low frequencies. Single-ended shield connection may be more advantageous when:

- an equipotential bonding conductor cannot be laid;
- analog signals (of a few mV or mA) are to be transmitted;
- foil (static) shields are used.

With data lines for serial communication, always use metal or metallized connectors. Secure the shield of the data line to the connector case. Do ***not*** connect the shield to PIN 1 of the connector.

For stationary operation, it is advisable to fully strip the insulation from the shielded cable and connect it to the shield/protective conductor bar.

Note

In the event of potential differences between ground points, a circulating current may flow through the shield connected at both ends. In this case, install an additional equipotential bonding conductor (see section 3.4.2).

Please observe the following points when connecting the shield:

- Use metal cable clamps for securing the braided shield. The clamps must enclose the shield over a large area and provide a good contact (see Figure 3-4).
- Connect the shield to a shield bar immediately after the cable entry into the cabinet. Route the shield as far as the module but do not connect it there again.

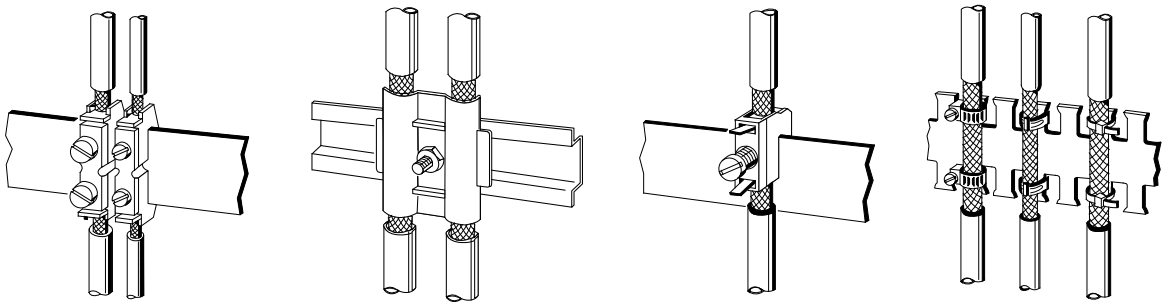


Figure 3-4. Examples of Securing Shielded Lines with Cable Clamps

3.4.4 Special Measures for Interference-Free Operation

Fitting Quenching Elements to Inductances

As a rule, inductances such as contactor or relay coils controlled by SIMATIC S5 do not require external quenching elements in the circuit, because the quenching elements are already integrated in the modules.

Inductances should be fitted with quenching elements only

- If SIMATIC S5 output circuits can be switched off by additionally fitted contacts, such as relay contacts for EMERGENCY OFF. In this case the integrated quenching elements in the modules are no longer effective;
- If they are ***not*** controlled by SIMATIC S5 modules.

You can place freewheel diodes, varistors or RC networks in circuit with inductances.

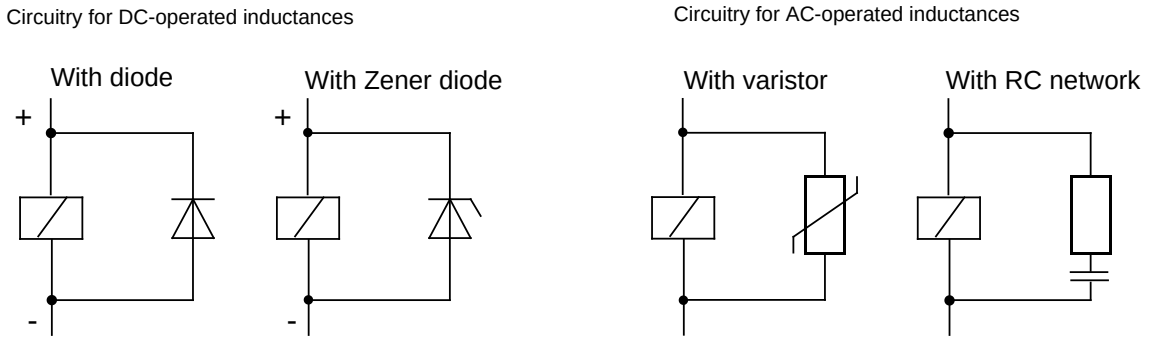


Figure 3-5. Quenching Circuits for Inductances

AC Power Connection for Programmers

A power socket should be fitted in each cabinet for the AC supply to programmers. The sockets should be powered from the distribution system to which the protective conductor for the cabinet is also connected.

Cabinet Lighting

Use incandescent bulbs, such as LINESTRA® lamps, for cabinet lighting. Avoid using fluorescent lamps because they generate interference fields. If the use of fluorescent lamps cannot be avoided, apply the measures shown in the following figure.

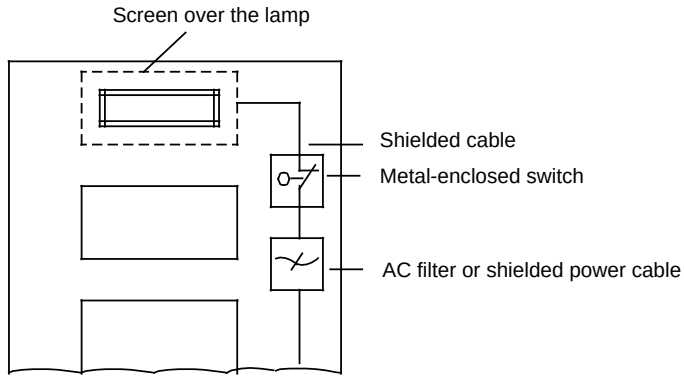


Figure 3-6. Measures for Interference Suppression of Fluorescent Lamps in a Cabinet

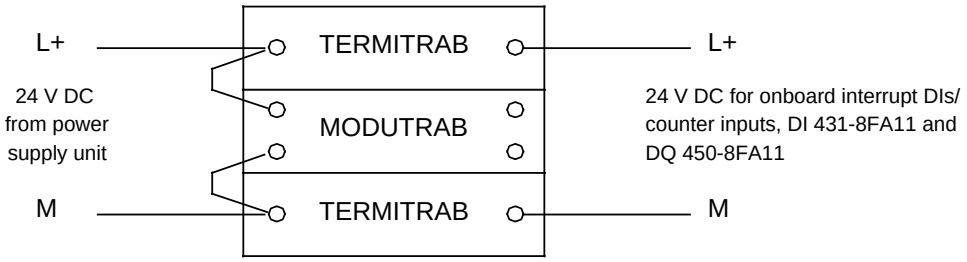
3.4.5 Filters for 24 V DC Power Supply Units

In order for the S5-95F to fully fulfill the IEC 801-5 (surge pulse) requirement for severity level 3, the 24 V DC supply for the following components must be filtered:

- Onboard interrupt DIs/counter inputs
- Digital input DI 431-8FA11
- Digital output DQ 450-8FA11

Filters for Protection to IEC 801-5, Severity III

Route the voltage generated by 24 V DC power supply units through a filter to the components listed above. Use the circuit arrangement shown in the Figure with the specified components.



To reduce interference, the lines between filter and S5-95F may not exceed a length of 50 cm.

Figure 3-7. Filters for Protection to IEC 801-5, Severity III

Table 3-3. Components for Network Filters

Component/Type	Manufacturer	Order No.
TERMITRAB Type SLKK-S/60 AC	PHÖNIX-CONTACT	27 94 974
MODUTRAB Type MT-2/1-S-24 DC	PHÖNIX-CONTACT	27 65 699

3.4.6 Checklist for the Electromagnetically Compatible Installation of Control Systems

Table 3-4. Checklist for Electromagnetically Compatible Installation

EMC Measures	Notes
Connection of inactive metal parts (see section 3.3.1) Check, in particular, the connections on: • Subracks • Cabinet members • Shield and protective conductor bars	
Do all inactive metal parts have a large-area, low-impedance interconnection and ground?	
Is there a satisfactory connection to the ground/protective conductor system?	
Have insulating layers on painted and anodized surfaces been removed, or have special contact washers been used for the connections?	
Are connections protected from corrosion, e.g. by grease?	
Are cabinet doors connected to the cabinet element with grounding strips?	
Routing of cables (see section 3.4.1)	
Cabling subdivided into line groups?	
Supply cables (230 to 400 V) and signal lines laid in separate ducts or bundles?	
Equipotential bonding (see section 3.4.2)	
With a separate arrangement, check that the equipotential bonding conductor has been correctly laid	
Shielding of cables (see section 3.4.3)	
Have metal connectors been used throughout?	
Are all analog and data lines shielded?	
Are line shields connected to the shield or protective conductor bar at the cabinet entry?	
Are line shields secured with cable clamps over a large area and at low impedance?	
Are line shields connected at both ends where possible?	
Inductances (see section 3.4.4)	
Are contactor coils which are not switched via SIMATIC contacts fitted with quenching elements?	
Filters for 24 V DC power supply units (see section 3.4.5)	
Was the supply voltage for onboard interrupt DIs/counter inputs, DI 431-8FA11 and DQ 450-8FA11 filtered?	

4 Installing and Connecting the Basic System

4.1	Basic System	4 - 1
4.1.1	S5-95F Basic Unit	4 - 2
4.1.2	Power Supply for the S5-95F Basic Unit	4 - 4
4.1.3	Setting the Subunit Identifier and the Length of the Fiber Optic Cable	4 - 5
4.2	Using the Onboard I/Os	4 - 7
4.2.1	Using the Failsafe, Onboard Digital Inputs	4 - 9
4.2.2	Short-Circuit Test for Sensor Lines	4 - 11
4.2.3	Using the Failsafe Onboard Digital Outputs	4 - 18
4.3	Connecting Actuators to Failsafe Digital Outputs	4 - 21
4.4	Onboard Interrupt Inputs	4 - 23
4.5	Onboard Counter Inputs for Counting and Monitoring Rotational Speed	4 - 23
4.5.1	Connection of Counter Inputs	4 - 24
4.5.2	Parameterizing Counter Inputs	4 - 25
4.5.3	Using Counter Inputs for Counting Tasks	4 - 26
4.5.4	Using Counter Inputs for Frequency Monitoring and Speed Monitoring	4 - 27
4.5.5	Scanning and Resetting the Counter Status	4 - 29
4.6	Failure and Monitoring of the Supply Voltages	4 - 30
4.7	Connector Pin Assignment of the Onboard I/Os	4 - 31

Figures

4-1 Basic System . . . . . 4 - 1
4-2 Mounting a Basic Unit on a Standard Mounting Rail . . . . . 4 - 2
4-3 Removing the Programmable Controller . . . . . 4 - 3
4-4 Connecting the Power Supply to the Basic Unit . . . . . 4 - 4
4-5 Position of the DIP Switch for Subunit Identifier and Fiber Optic Cable Length . . . . . 4 - 5
4-6 Setting the Subunit Identifiers . . . . . 4 - 6
4-7 Setting the Length of the Fiber Optic Cable . . . . . 4 - 6
4-8 Connection of a Single-Channel, Failsafe Sensor . . . . . 4 - 9
4-9 Connection of a Two-Channel Sensor . . . . . 4 - 10
4-10 Short-Circuit Test from the Viewpoint of the Operating System . . . . . 4 - 12
4-11 Connection of a Check DQ for Short-Circuit Test for Single-Channel, Failsafe Sensor . . . . . 4 - 15
4-12 Connection of a Check DQ for Short-Circuit Test, Two-Channel Sensor . . . 4 - 16
4-13 Connection of Two Check DQs for Short-Circuit Test, Two-Channel Sensor . . . . . 4 - 17
4-14 Connection of a Failsafe Actuator . . . . . 4 - 19
4-15 Indirect Connection of an Actuator via Coupling Elements . . . . . 4 - 20
4-16 Connecting a Counter Input . . . . . 4 - 24

Tables

4-1 Mounting, Removing and Exchanging the Hardware . . . . . 4 - 3
4-2 Reactions of Signal Groups . . . . . 4 - 7
4-3 Discrepancy Times for Digital Inputs . . . . . 4 - 8
4-4 Wiring Diagram for Short-Circuit Test . . . . . 4 - 13
4-5 Terminal Assignment for Onboard DQ . . . . . 4 - 18
4-6 Blanking Times for Testing Digital Output Modules . . . . . 4 - 21
4-7 Permissible Actuators for Connection to the S5-95F . . . . . 4 - 21
4-8 Resetting the Diagnostic Byte in the Program after a Counter Overflow . . . 4 - 26
4-9 Comparison Values for Frequency Monitoring . . . . . 4 - 27
4-10 Response Times for Rotational Speed Monitoring . . . . . 4 - 28
4-11 Addresses and Counter Access Possibilities . . . . . 4 - 29
4-12 Monitoring the Supply Voltages for Onboard I/Os . . . . . 4 - 30
4-13 Assignment of the 40-Pin I/O Connector (Left Side of Connector) . . . . . 4 - 31
4-14 Assignment of the 40-Pin I/O Connector (Right Side of Connector) . . . . . 4 - 32
4-15 Assignment of the 9-Pin Sub D Connector . . . . . 4 - 32

4 Installing and Connecting the Basic System

This section contains the following information:

- Components required for a basic system
- Mounting of the devices
- Wiring of the onboard I/Os

4.1 Basic System

The S5-95F basic system is always of redundant design and consists of two basic units, which are interconnected via a fiber optic link.
Both basic units are of identical construction and comprise the CPU, an internal power supply and onboard I/Os.

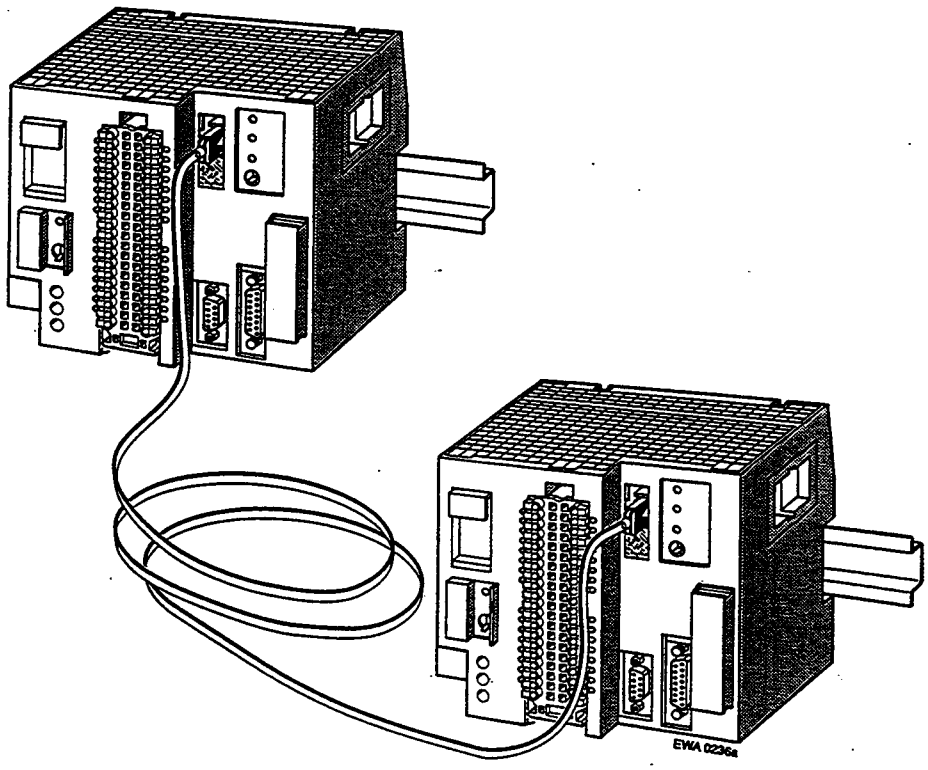


Figure 4-1. Basic System

The following components are required for a basic system:

- Two S5-95F basic units (same version)
- A fiber optic cable
- At least one electrically separated 24 V DC power supply

4.1.1 S5-95F Basic Unit

As shown in Figure 4-1, an S5-95F programmable controller consists of two basic units. The following section describes mounting of one basic unit. The other basic unit is mounted in the same way.

How to Mount an S5-95F Basic Unit

The basic unit is mounted on a standard mounting rail:

- Hook the basic unit onto the standard mounting rail.
- Swing it back, until the slide on the bottom of the basic unit audibly snaps into place.

Figure 4-2 shows the installation of a basic unit.

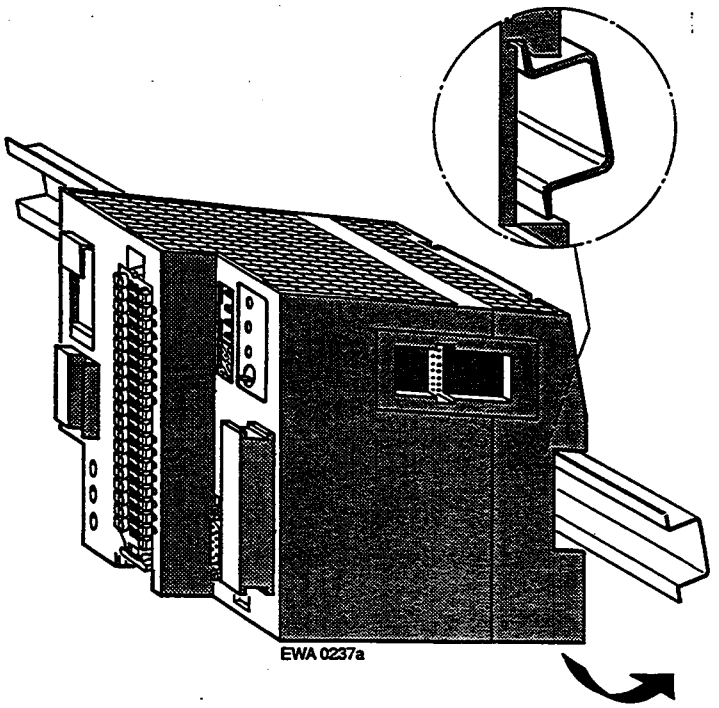


Figure 4-2. Mounting a Basic Unit on a Standard Mounting Rail

Removing of the basic unit:

- Remove all connected supply cables and signal cables.
- Using a screwdriver, press down on the slide on the bottom of the controller.
- Swing the basic unit up and out of the standard mounting rail.

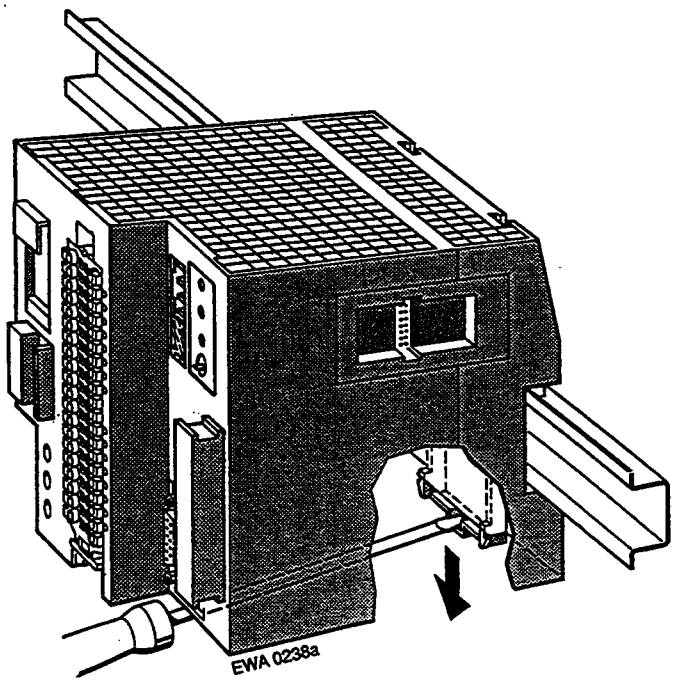


Figure 4-3. Removing the Programmable Controller

Use the following table when you install, remove, or change the programmable controller installation:

Table 4-1. Mounting, Removing and Exchanging the Hardware

Mounting, Removing, and Modifying:	Status of Power Supply	PLC Operating Mode	Load Voltage
Basic unit	Supply voltage OFF	Not relevant	Not relevant
Bus units Interface modules 40-pin on-board I/O connector 9-pin sub D connector for interrupt/counter inputs	POWER OFF	Not relevant	Not relevant
I/O modules	Not relevant	STOP/RUN*	Not relevant

* When I/O modules are plugged in or removed in the RUN mode, the S5-95F disables the outputs for approx. 300 ms. Furthermore, handling of the I/O modules can result in voltage dips, which cause the S5-95F to switch to the STOP mode.

4.1.2 Power Supply for the S5-95F Basic Unit

The S5-95F must be operated with an electrically isolated power supply. The rated voltage is 24 V DC.

24 V DC Power Supply for the S5-95F

If your automation system is made up of basic units 095-8FB01 you can use the same power supply for both basic units. This is possible because basic units 095-8FB01 have high-grade internal facilities for monitoring the 24 V DC voltage supply.

Note that in the event of an impermissible voltage rise (V > 33 V DC) a fuse in the basic unit will blow. This fuse is installed for safety reasons and you must not replace it yourself under any circumstances.

24 V DC power supplies must have the following characteristics:

- Safe (electrical) isolation to EN 60950 or safe (electrical) isolation to DIN VDE 0551 or EN 60742 and DIN VDE 0160.
  For special applications, it might be necessary to observe additional requirements (see rules and regulations in Chapter 1)
- 24 V DC rated voltage
  - Permissible tolerance: 20 V to 30 V (including ripple)
- Mains buffering > 20 ms

We recommend you to use the power supplies of our SITOP range in the S5-95F.

How to Connect the Power Supply to the S5-95F Basic Unit

- ▸ Connect terminals L+ and M of the power supply to the corresponding terminals of the basic unit (see Figure 4-4).
- ▸ Connect terminal 5 of the S5-95F basic unit with the standard mounting rail.

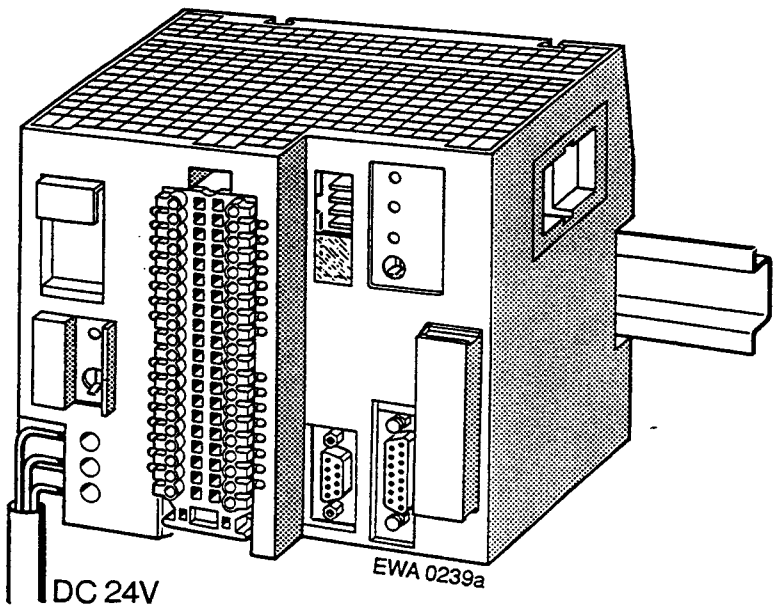


Figure 4-4. Connecting the Power Supply to the Basic Unit

4.1.3 Setting the Subunit Identifier and the Length of the Fiber Optic Cable

With the quadruple DIP switch below the connector for the fiber optic cable you set the following:

- Subunit identifier
- Length of fiber optic cable

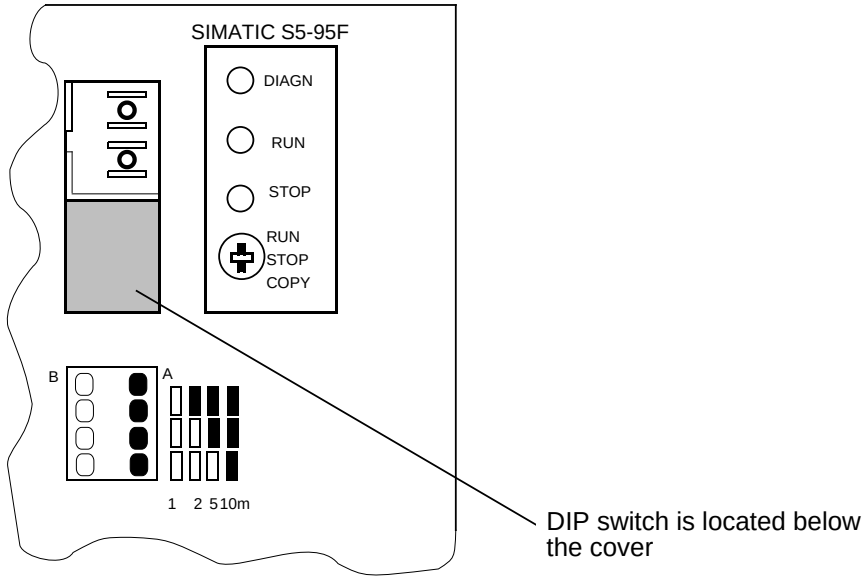
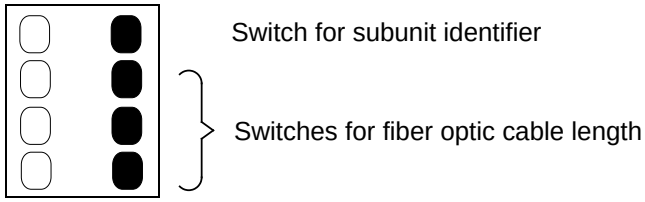


Figure 4-5. Position of the DIP Switch for Subunit Identifier and Fiber Optic Cable Length

DIP Switch Assignment



How to Set the Subunit Identifier

Set the subunit identifier on both subunits.
One subunit is assigned identifier "A", the other identifier "B".



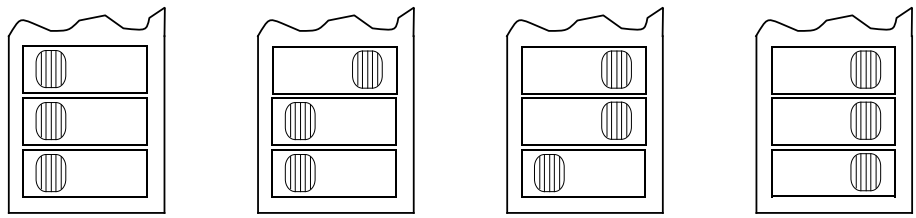
Subunit A Subunit B

Figure 4-6. Setting the Subunit Identifiers

You should observe that by setting the subunit identifers you also define the addressing and wiring rules for the I/Os.

How to Set the Length of the Fiber Optic Cable

Use the lower three switches to set the length of the fiber optic cable. The fiber optic cable can be 1, 2, 5 or 10 m long (3.28, 6.56, 16.4 or 32.8 ft).



1 m (3.28 ft) 2 m (6.56 ft) 5 m (16.4 ft) 10 m (32.8 ft)

Figure 4-7. Setting the Length of the Fiber Optic Cable

How to Connect the Fiber Optic Cable

Connect the two basic units via the fiber optic cable.

Note

Observe the following when connecting the fiber optic cable:

- Make sure that the connectors are not contaminated by dust, grease, etc., as this causes impermissible damping.
- The bending radii for fiber optic cables must at least be 40 mm.

4.2 Using the Onboard I/Os

The S5-95F offers you the following I/Os on the two basic units:

- 16 failsafe digital inputs (DI 32.0 to 33.7)
- 8 failsafe digital outputs (DQ 32.0 to 32.7)
- 8 non-failsafe digital outputs (DQ 33.0 to 33.3 in subunit A and DQ 32.0 to 34.3 in subunit B)
- 4 failsafe interrupt inputs (DI 59.0 to DI 59.3)
- 2 failsafe counter inputs (IW 36 for counter A, IW 38 for counter B)

The addresses of the inputs and outputs are permanently assigned and cannot be modified.

Prior to using the onboard I/Os, you must

Wire the corresponding input and output
Assign the input and output parameters using COM 95F
Assign certain signal groups to the inputs and outputs.

A summary of the assignment of the I/O connectors is given in section 4.7.

Signal Groups

A signal group is a combination of I/Os for which a uniform system reaction is defined.
The reaction to a fault within a signal group is set using COM 95F (see COM 95F manual).
The assignment of the signal groups applies not only to the onboard I/Os, but also to the failsafe external I/Os.

Table 4-2. Reactions of Signal Groups

Parameterizable Reaction for Signal Group	Remarks
STOP	As soon as the S5-95F recognizes an I/O fault, it switches the whole system to STOP
Passivation	As soon as the S5-95F recognizes an I/O fault, it passivates all I/Os of the signal group affected: Inputs are read with "0" signal Outputs are reset
User reaction AND OR OLD	If an I/O fault occurs, the S5-95F signals this fault, and reads the faulty input in the case of AND with "0" signal OR with "1" signal OLD with the last value valid (old value)

Discrepancy Time

Usually, the read result for an input signal is identical in both subunits. However, in certain situations, there can sometimes be a discrepancy. Reasons for this may be:

- A hardware failure, such as failure of a sensor or input module
- A difference in the instant of access of the two subunits
- Different switching instants in the case of two-channel sensors/contacts

In order to differentiate between a hardware failure and a fleeting, coincidental signal change, the safety-related input signals are subjected to a discrepancy analysis.The analysis is executed by the operating system; you simply specify a discrepancy time in DB1, within which the signals match again.

Safety Note

The discrepancy times must be selected so that the sum of discrepancy time and total PLC response time is less than the error tolerance time of the relevant process function.

Table 4-3. Discrepancy Times for Digital Inputs

Type of Input	Address	Discrepancy Time Programmable in DB1
Hardware interrupt DI	IB 59	Discrepancy time, short : approx.1 ms
Software interrupt DI	IB 32 to 33, IB 59	Discrepancy time, short : approx.1 ms Discrepancy time, average : approx. 5 ms Discrepancy time in : n* OB1 cycles
Standard DI	IB 0,2 to 30	Discrepancy time in : n* OB1 cycles Discrepancy time in : n* OB13 cycles
	IB 32 to 33	Discrepancy time, short : approx. 1 ms Discrepancy time in : n* OB1 cycles Discrepancy time in : n* OB13 cycles
	IB 59	Discrepancy time, short : approx. 1 ms

Note

- The average discrepancy time (approx. 5 ms) fluctuates with the synchronization interval. For systems with no load it is less than or equal to 5 ms, for systems with maximum interrupt load up to 15 ms.
- If you select the discrepancy time as a number of OB13 cycles, you must initialize the OB13 interval in DB1. However, you need not include the OB13 in the user program.

4.2.1 Using the Failsafe, Onboard Digital Inputs

The onboard digital inputs are located on the right-hand side of the 40-pin connector. They are galvanically isolated from the internal reference potential of the basic unit.

How to Connect the Onboard DI

Connect the 24 V DC load voltage (L+ potential to terminal 1, L- potential to terminal 20).
Connect the sensor to the selected digital input. You may use either two-channel or single-channel failsafe sensors.

Connection of Single-Channel, Failsafe Sensors (Type B)

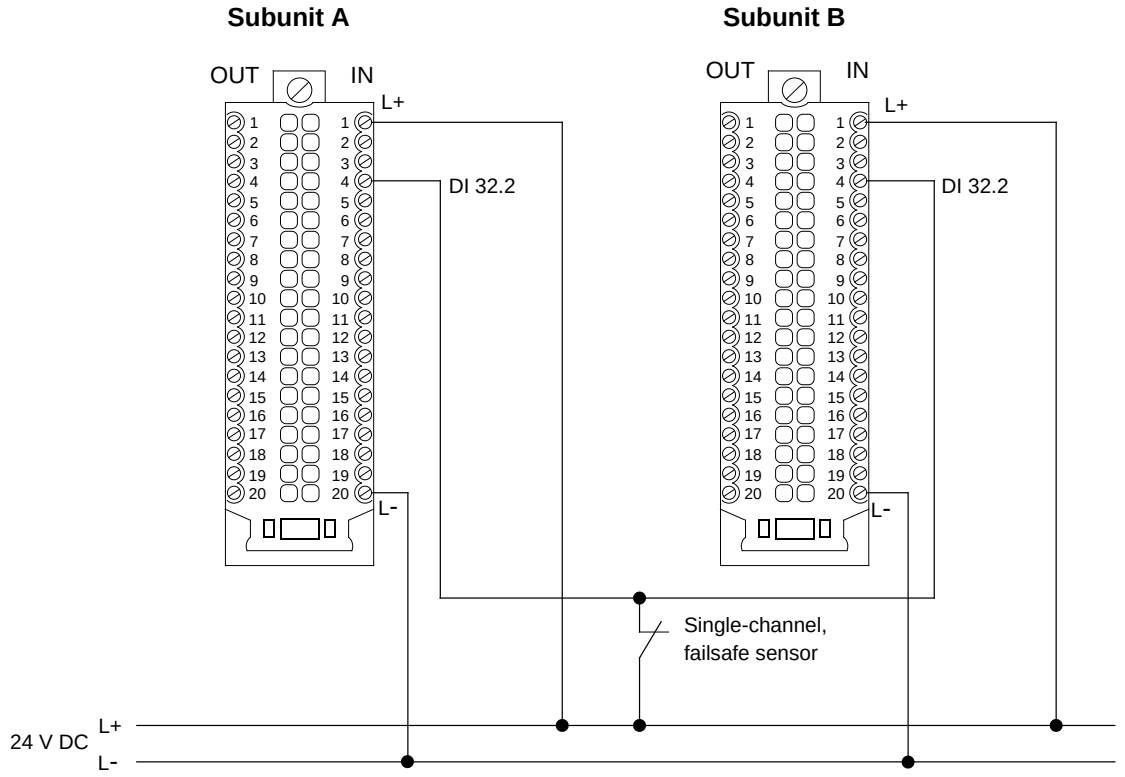


Figure 4-8. Connection of a Single-Channel, Failsafe Sensor

Connection of Two-Channel Sensors (Type C)

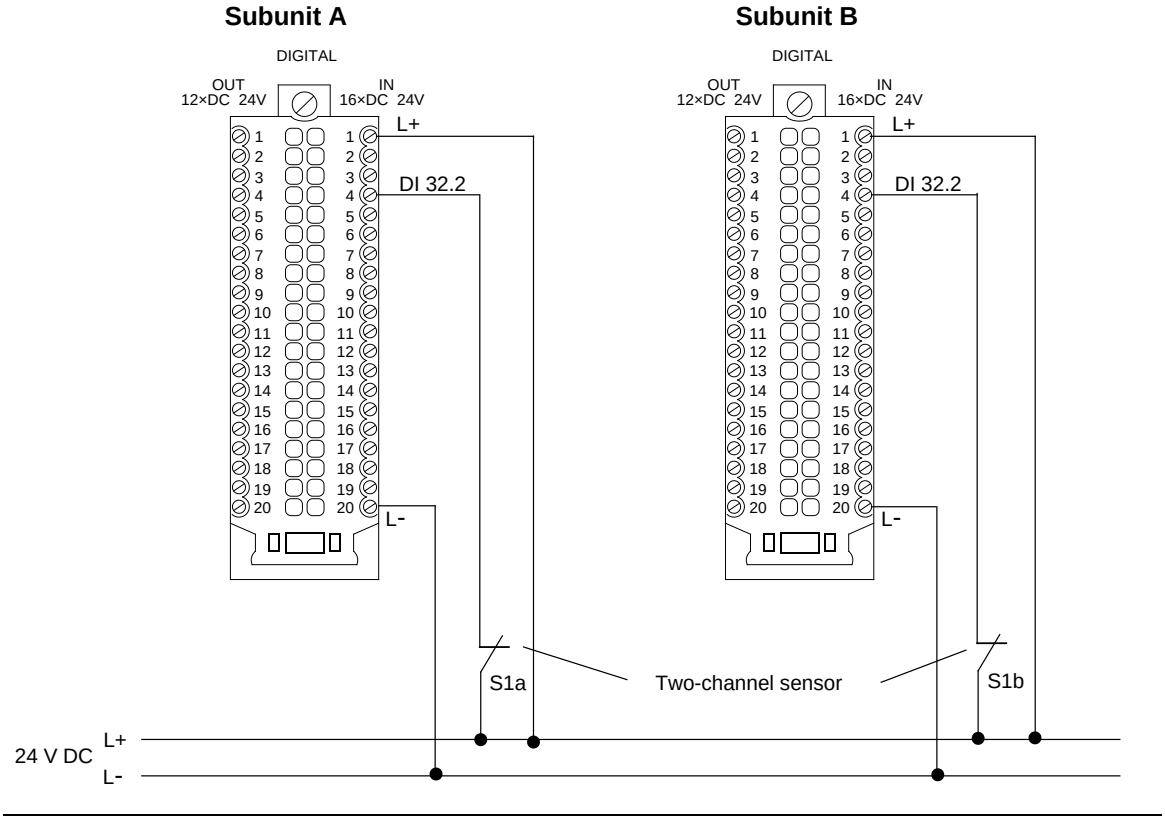


Figure 4-9. Connection of a Two-Channel Sensor

Requirements on Sensors for Failsafe Digital Inputs

Please refer to section 18.8.

How to Program Onboard Digital Inputs in DB1

The parameters of the onboard DIs are programmed in DB1. In the basic setting (default) all failsafe onboard DIs are assigned short discrepancy times. You can change this parameter in DB 1 with COM 95F

The discrepancy time of the onboard DIs can be specified as

- Short discrepancy time (approx. 1 ms)
- Discrepancy time in n OB1 cycles
- Discrepancy time in n OB13 cycles

4.2.2 Short-Circuit Test for Sensor Lines

When integral function block FB 252 is properly initialized with COM95 and called, the S5-95F also monitors the sensor lines to failsafe onboard and external digital inputs. This monitoring procedure relates to a short-circuit between two sensor lines and can be carried out for sensor lines that are connected to different check DQs (Q33.0 to Q33.3 and Q34.0 to Q34.3).

With one check DQ you can monitor up to 8 sensor lines simultaneously, no matter whether the sensor lines lead to onboard inputs or to external inputs. If you connect several sensor lines to one check DQ, a conductor-to-conductor short circuit between these lines must be physically excluded (e.g. lines laid separately and/or with double insulation).

Note

To monitor the sensor lines to failsafe onboard DIs of IB 59, only onboard DIs of IB 59 may be connected to the check DQ used for this purpose.
The maxing of IB 59 onboard inputs and other failsafe DIs on the same check DQ is not permitted.

Monitoring is optional and may be necessary when

- a short-circuit between commonly run sensor lines could occur, e.g. when several signal lines are run in a single mechanically stressed spiral cable, or
- the process requires diagnostics for the connected sensor lines.

A short-circuit between two signal lines results in

- falsification of the sensor signal in the case of single-channel sensors, and
- loss of sensor redundancy in the case of two-channel sensors.

Method of Operation

The operating system monitors sensor lines that are connected to different check DQs. To do so, the operating system resets the check DQs (Q33.0 to A33.3 and Q34.0 to Q34.3) successively for a short time and evaluates the digital inputs.

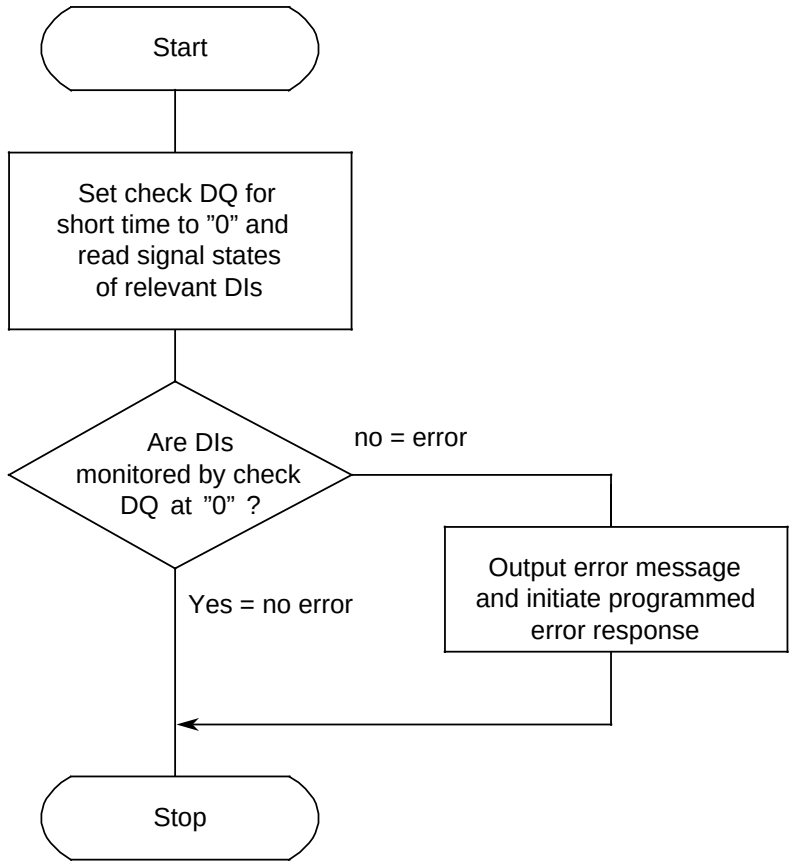


Figure 4-10. Short-Circuit Test from the Viewpoint of the Operating System

You can initiate testing of the sensor lines whenever you want to by invoking FB252 (see Chapter 9).

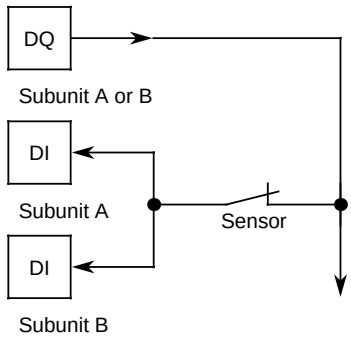
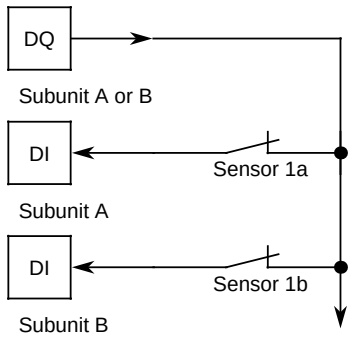
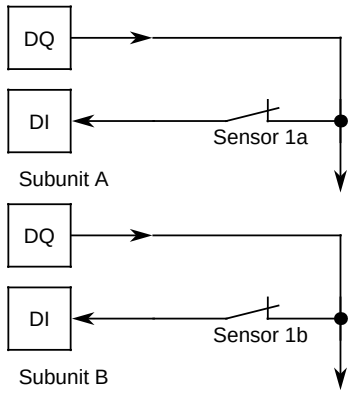
Response to Failure of the Check DQ

If a short circuit to ground (reference potential) occurs at a check DQ, the S5-95F automatically disables the output without reporting an error (LED on the module, however, remains on).

The check DQ remains disabled until
- the short-circuit has been eliminated and
- the output has been reset in the user program (for instance by calling the short-circuit test via FB252).

Wiring Diagrams for Short-Circuit Test

Table 4-4. Wiring Diagram for Short-Circuit Test

Wiring Diagram	Description
I/O type D 	Line monitoring for failsafe DI with single-channel failsafe sensor. Check DQ tests DI in subunit A and subunit B.
I/O type E 	Line monitoring for failsafe DI with two-channel sensor. Check DQ tests DI in subunit A and subunit B. Application: Sensor lines are laid so as to be ***inherently short-circuit-proof*** (e.g. in separate non-metallic-sheathed cable).
I/O type F (from 095-8FB01) 	Line monitoring for failsafe DI with two-channel sensor. Check DQ in subunit A tests DI in subunit A. Check DQ in subunit B tests DI in subunit B. Application: Sensor lines are in a common non-metallic-sheathed cable or in a conductor bundle and ***must not be laid so as to be inherently short-circuit-proof***. Note the special parameters for line monitoring (possible with 095-8FB01 base units only).

There are no restrictions on which check DQ can be used for I/O types D, E and F. However, please note that I/O type F requires two check DQs, which, of course, you can no longer use for I/O types D and E.

Initializing the Parameters for Line Monitoring

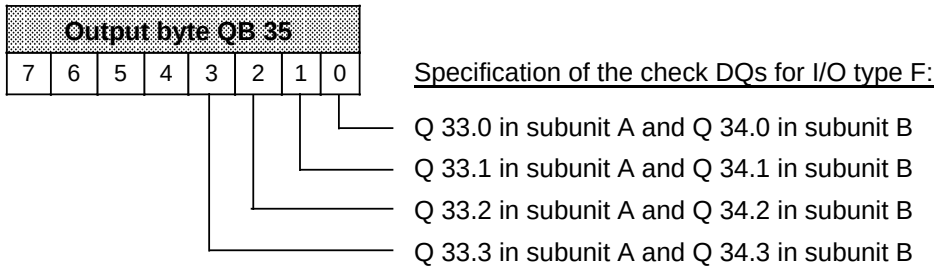
In addition to the appropriate wiring, parameters must also be assigned for line monitoring using COM 95F. When initializing the I/Os, each failsafe DI must be assigned to the associated check DQ (see COM 95F manual).

Initializing the parameters for I/O types D and E:

If you want to use I/O type D or E in conjunction with line monitoring, you must specify the check DQ used with COM 95F (see COM 95F manual).

Exception for I/O type F:

If you want to use I/O type F in conjunction with line monitoring, you must
- set the corresponding bit in QB 35 in the start OB (OB 21, OB 22) (see Table)
- specify the check DQ used in subunit A in COM 95F.



Invoking the Line Monitoring Test

You must start line monitoring in your user program by calling a test routine. Start this test routine with FB252. Please read the description of FB252 in Chapter 9.2.9.

How to Connect Check DQs

Connect the 24 V DC load voltage for the check DQs
(L+ potential to terminal 11 and L- potential to terminal 20)
The load voltage source must also supply the failsafe DIs.
Connect the sensor lines which are to be monitored for conductor-to-conductor short circuit to different check DQs.

Monitoring the Connection of Failsafe Onboard DI with Single-Channel, Failsafe Sensor via Check DQ (Type D)

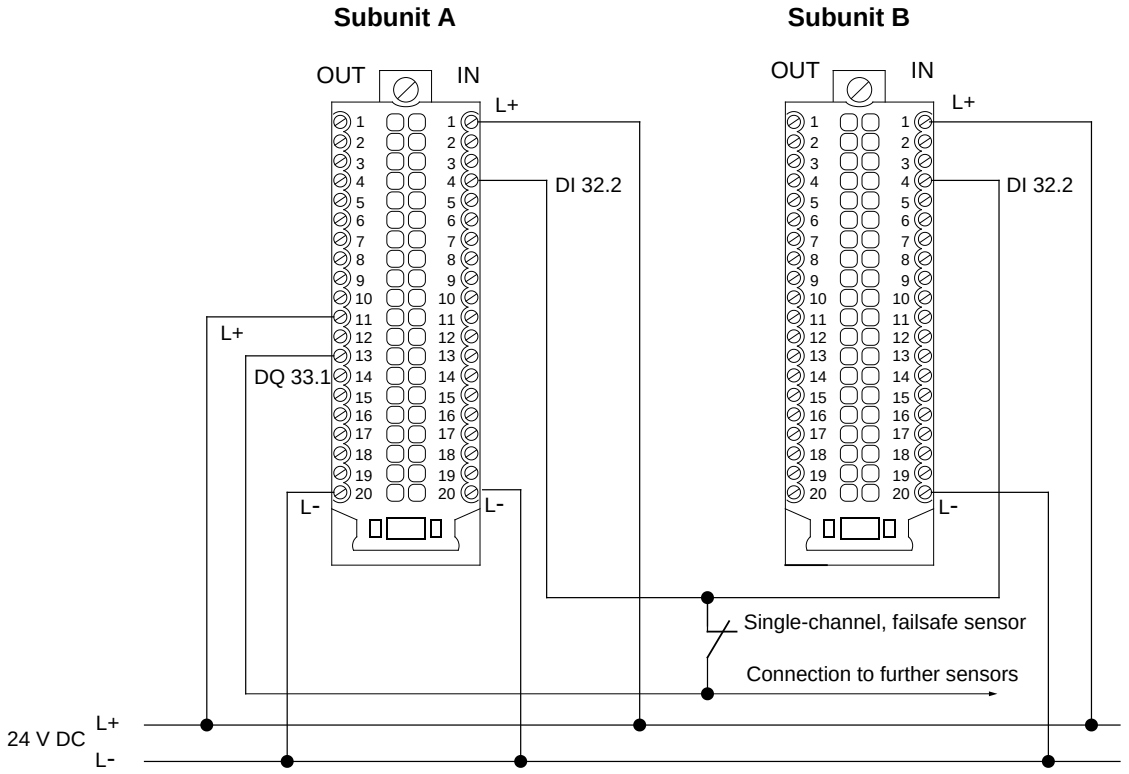


Figure 4-11. Connection of a Check DQ for Short-Circuit Test for Single-Channel, Failsafe Sensor

Monitoring the Connection of Failsafe Onboard DI with Two-Channel Sensor via Check DQ (Type E)

A conductor-to-conductor short circuit between the sensor lines of S1a and S1b can (physically) not occur, since both lines are laid separately. Therefore, both sensor lines can be monitored by the same check DQ.

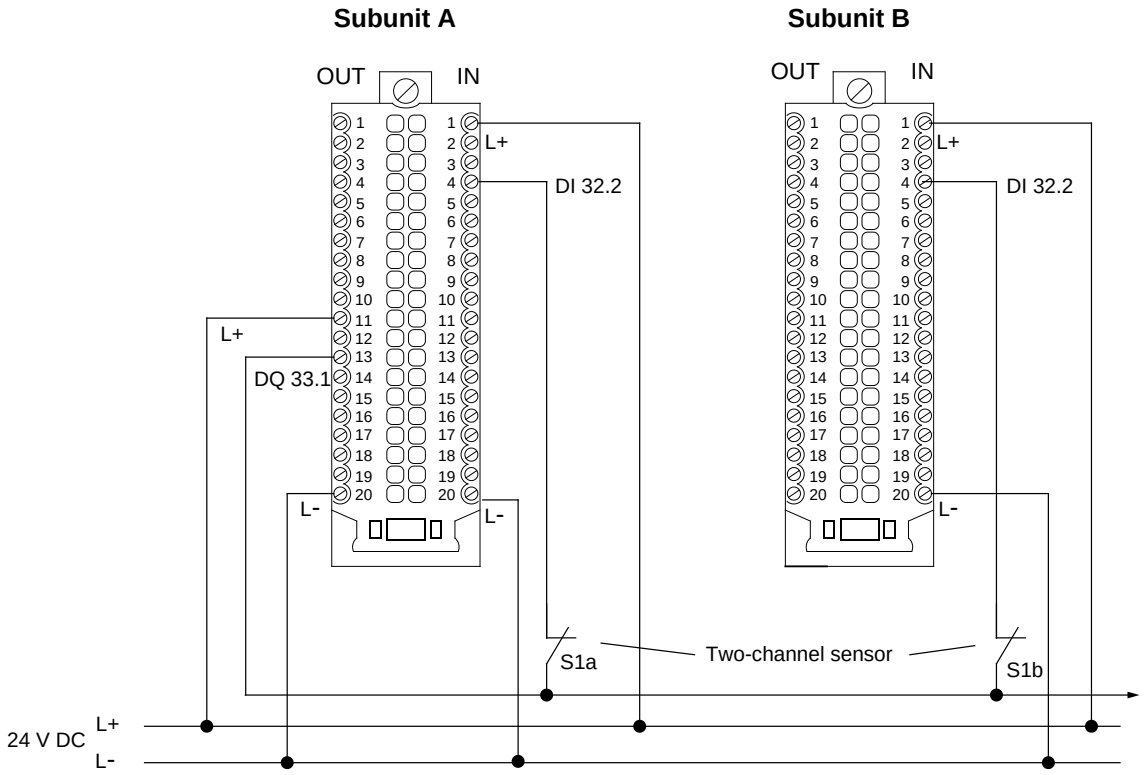


Figure 4-12. Connection of a Check DQ for Short-Circuit Test, Two-Channel Sensor

Requirements on Sensors for Failsafe Digital Inputs

Please refer to section 18.8.

How to Program Check DQs for Short-Circuit Test in DB1

In DB1, you program which digital inputs you supply via check DQs.

Monitoring the Connection of Failsafe Onboard DIs with Two-Channel Sensor via Check DQ (Type F)

A **short-circuit is possible** between sensor lines of S1a and S1b because both lines are laid in e.g. a non-metallic sheathed cable. For this reason, the sensor lines must be monitored by different check DQs.

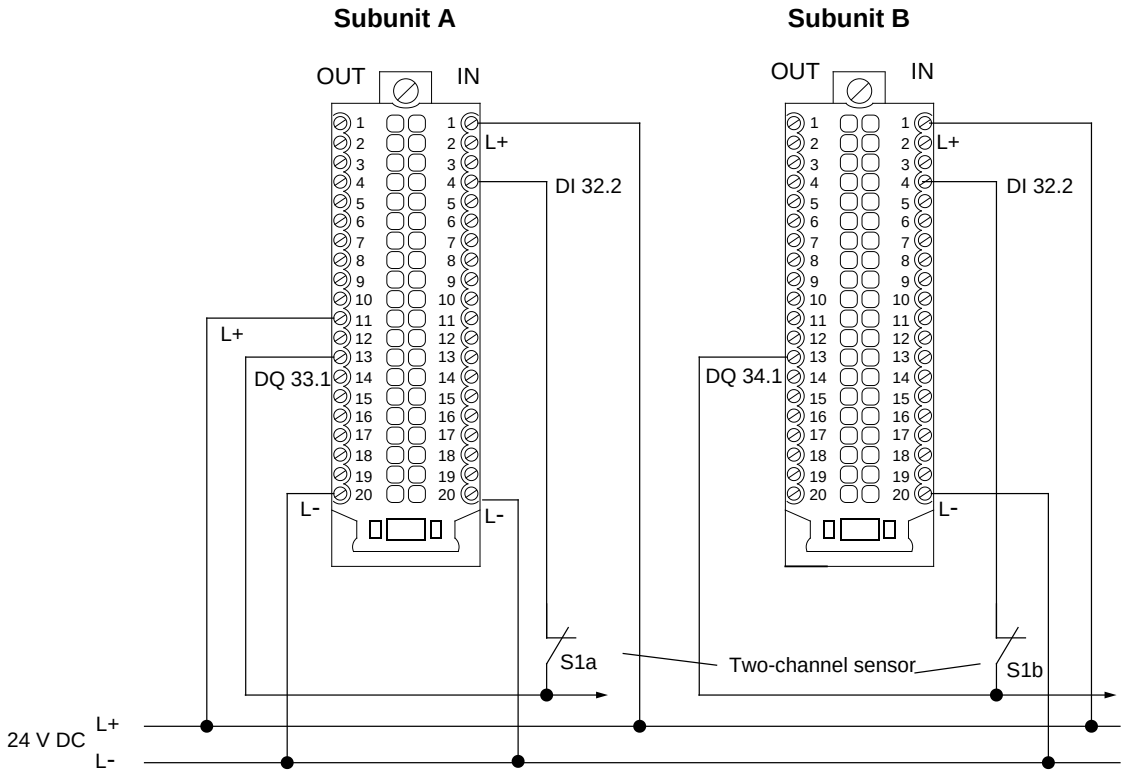


Figure 4-13. Connection of Two Check DQs for Short-Circuit Test, Two-Channel Sensor

Requirements on Sensors for Failsafe Digital Inputs

Please refer to section 18.8.

How to Program Check DQs for Short-Circuit Test

- In DB1, specify the digital inputs initialized via check DQs from subunit A.
- In the start OBs (OB 21/22), specify in output byte QB 35 the check DQ pair used for I/O type F.

4.2.3 Using the Failsafe Onboard Digital Outputs

The onboard DQs are located on the left-hand side of the 40-pin connector. They are galvanically isolated from the internal reference potential of the basic unit.

When connecting the DQ, make sure that one bit is assigned different terminals in subunit "A" and "B". For connector pin assignment please refer to section 4.7.

Table 4-5. Terminal Assignment for Onboard DQ

DQ Channel	Subunit A		Subunit B	
	Terminal	Output Potential	Terminal	Output Potential
Q 32.0	2	L+	6	L-
Q 32.1	3	L+	7	L-
Q 32.2	4	L+	8	L-
Q 32.3	5	L+	9	L-
Q 32.4	6	L-	2	L+
Q 32.5	7	L-	3	L+
Q 32.6	8	L-	4	L+
Q 32.7	9	L-	5	L+

How to Program the Onboard DQ in DB1

You program the corresponding signal group for the failsafe outputs in DB1. The basic setting (default) for the onboard DQ byte is signal group 0.

Example: Failsafe Connection of an Actuator with Direct Triggering (Type K)

Connect the 24 V DC load voltage (L+ potential to terminal 1, L- potential to terminal 10).
With direct triggering, the load voltage for subunit A and for subunit B must be supplied by one voltage source.
Connect the actuator to the output selected.
The actuator receives the L+ potential from the first subunit and the L- potential from the other subunit (P-M triggering). When connecting the DQ, make sure that one bit is assigned different terminals in "A" and "B".

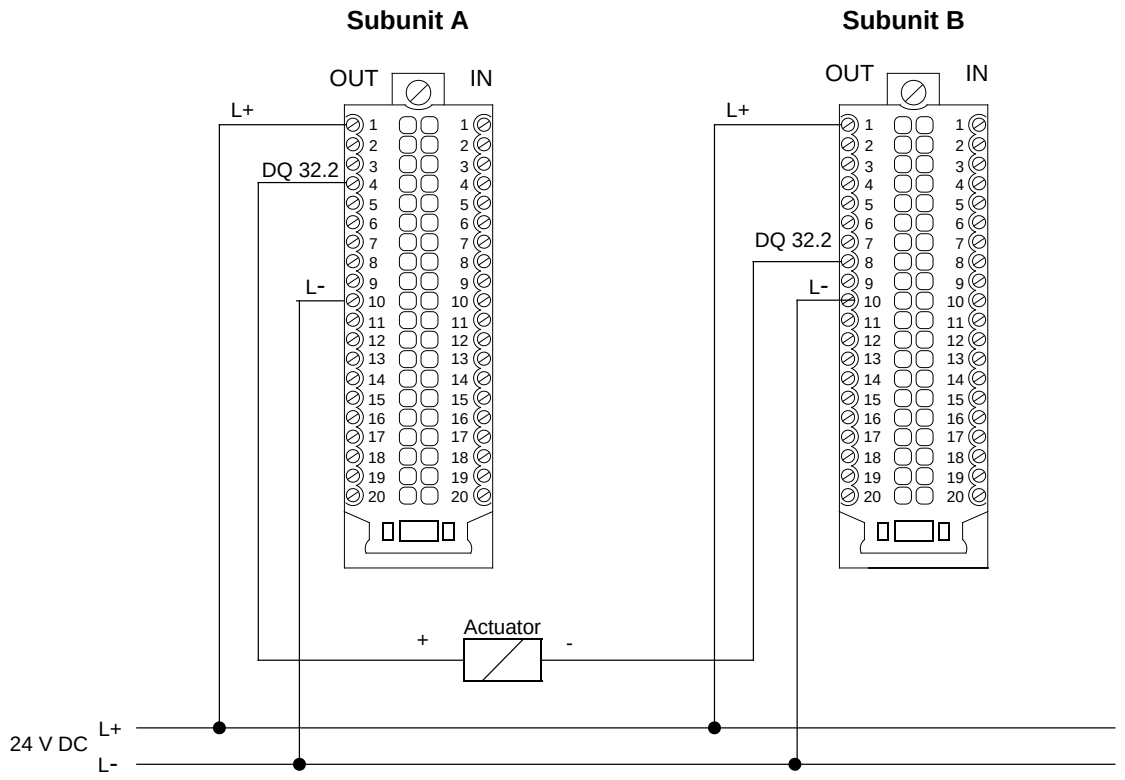


Figure 4-14. Connection of a Failsafe Actuator

Example: Indirect Connection of Actuators via Coupling Elements (Type L)

Connect the 24 V DC load voltage (L+ potential to terminal 1, L- potential to terminal 10).
Connect the coupling relays to the outputs selected (see Figure 4-15).
Please note, that some applications (e.g. special valves for burner controls) require the use of diverse coupling relays. A list of suitable actuators can be found in section 4.3.

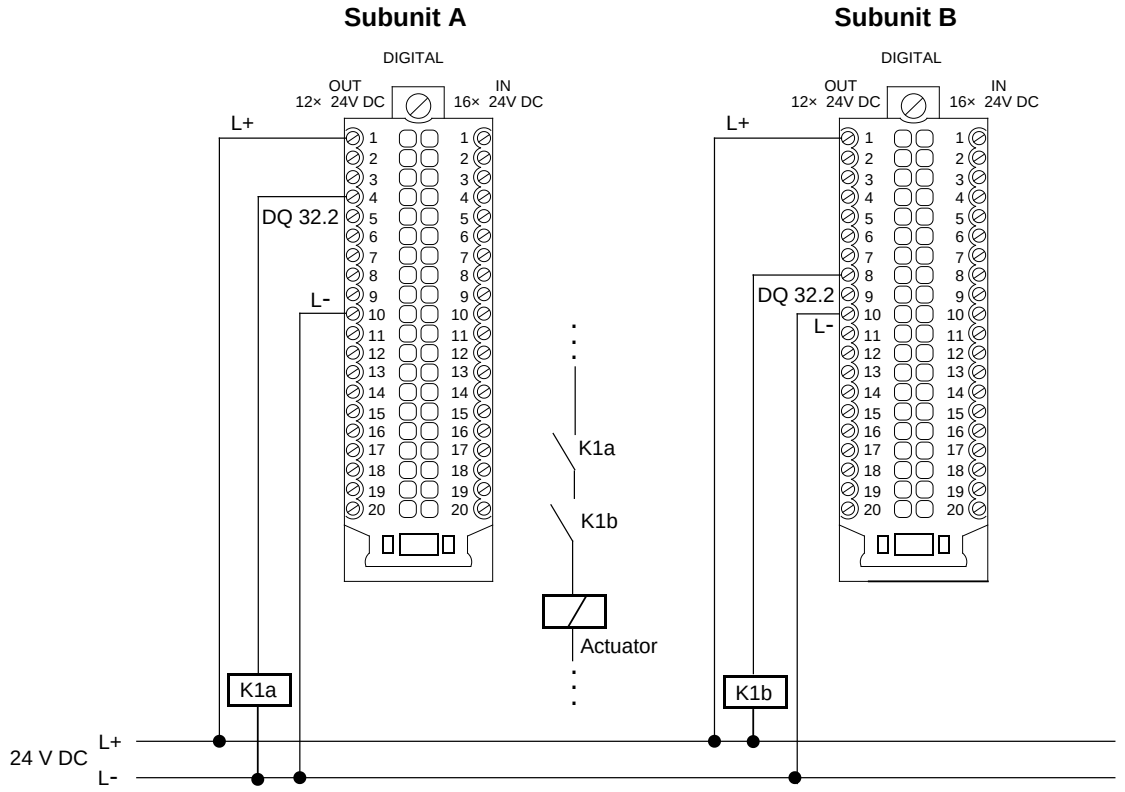


Figure 4-15. Indirect Connection of an Actuator via Coupling Elements

4.3 Connecting Actuators to Failsafe Digital Outputs

In RUN, the S5-95F tests the digital output modules once every hour. To do so, it disables the outputs for a brief period of time, the length of which is different for onboard I/Os than for external I/Os (blanking time).

Table 4-6. Blanking Times for Testing Digital Output Modules

I/Os Tested by the S5-95F	Blanking Time at Output
Onboard digital outputs	< 1 ms
DQ 450-8FA11 external digital output modules	< 7 ms
DQ 450-8FA12 external digital output modules	< 1 ms

With the exception of the (shorter) blanking time, the DQ 450-8FA12 module is functionally fully upward-compatible with the DQ 450-8FA11. For this reason, it is possible to drive an actuator via a module pair consisting of one DQ 450-8FA11 and one DQ 450-8FA12. Note, however, that in this case the driven actuator must be oriented to the DQ 450-8FA11's longer blanking time.

Fast-reacting actuators (e.g. quick-acting valves), may temporarily drop out during the test. If your process does not tolerate this, you should select one of the following three options:

- Use actuators with a sufficient lag (see Table 4.7)
- Call the DQ test at a non-critical point in the user program
- Connect diodes and/or RC networks to the actuators

Using Actuators with Sufficient Lag

The following table shows a selection of tested coupling relays, which have a sufficient lag and which do not drop out during the DQ test.

Table 4-7. Permissible Actuators for Connection to the S5-95F

Manufacturer	Type	Contacts/ Rating
SIEMENS	3TF2010-0BB4	3×9 A at 400 V AC
SIEMENS	3TF4222-0BB4	4×10 A at 230 V AC
SIEMENS	3TF4322-0BB4	4×10 A at 230 V AC
SIEMENS	3TH4262-0BB4	8×10 A at 230 V AC
SIEMENS	3TH4382-0BB4	10×10 A at 230 V AC
SIEMENS	3TH8031-0B	4×10 A at 230 V AC
SIEMENS	3TH3022-0B	4×10 A at 230 V AC
SIEMENS	3TH2022-0BB4	4×4 A at 230 V AC
Télémécanique	LP4-EC09	4×6 A at 230 V AC
ABB	KC22E	4×4 A at 230 V AC
AEG	SH04	4×6 A at 440 V AC

Calling the DQ Test Only in the User Program

Since the outputs are briefly disabled during the DQ test (see Table 4.6), it is possible that fast-reacting actuators drop out unintentionally during operation.You should therefore ***not*** have the DQ test carried out ***automatically by the S5-95F*** (background test), but at defined, uncritical system states via the FB252 (see section 9.2.9).

Connecting Diodes and/or RC Networks to the Actuators

Connection of diodes and/or RC networks results in the actuators being released with a time delay. Dimensioning of the components required depends on the actuator used. If you have special questions, please contact your local Siemens representative.

4.4 Onboard Interrupt Inputs

There are two different types of onboard interrupts on the S5-95F:

- Onboard hardware interrupts (OB2 interrupts)
- Onboard software interrupts (OB3 interrupts)

Connection and parameterization of the interrupt inputs are described in Chapter 12 ("Interrupt Processing").

Supply Voltage for Interrupt Inputs (9-Pin Sub D Connector)

Please refer to section 4.6.

4.5 Onboard Counter Inputs for Counting and Monitoring Rotational Speed

The S5-95F is equipped with two separate counter inputs which can be used for counting pulses and/or for frequency monitoring/monitoring rotational speed. You must assign parameters to these inputs with COM 95F in order to be able to use them.

Each counter input is internally assigned a 16-bit counter register. The counter registers count the pulses that are received at the enabled counter inputs. When a comparison value, which can be preset in DB1, is reached, the S5-95F calls OB3.

In the following text, we differentiate between counter inputs and counter registers only when necessary. Otherwise, we briefly refer to counter A and counter B.

Supply Voltage for Counter and Interrupt Inputs (9-Pin Sub D Connector)

The supply voltage at the 9-pin Sub D connector and the counter inputs' signal voltage must be the same (24 V DC). For this reason, use the same voltage source for the 9-pin Sub D connector and the pulse generator (see Figure 4.16).

Also refer to the information concerning the monitoring of supply voltages presented in section 4.6.

Sensor Requirements for Onboard Counters

The following sensors are permissible for onboard counters:
- Single-channel, failsafe sensors
- Two-channel sensors, with contacts operating in synchronism so that the same switching states are possible within approx. 1 ms for all frequencies occurring during operation.

Signal Duration for Counting Pulses

To ensure that the S5-95F will recognize a counting pulse, signal levels "0" and "1" must always have a duration of at least 0.5 ms (sensor contacts may not bounce).

Maximum Operating Frequency and Permissible Frequency Changes

The counter inputs are 24 V DC inputs. The maximum operating frequency depends on the supply voltage for counter inputs and counter sensors.

Supply Voltage for Counters and Counter Sensors	Maximum Operating Frequency
20 V DC to 23 V DC	750 Hz
23 V DC to 30 V DC	900 Hz

Safety Note

If the counting signal exceeds the maximum operating frequency given above, it is possible that the S5-95F will no longer recognize pulses.
Note that, in this case, counting or failsafe rotational speed monitoring with the S5-95F is no longer reliable.

The frequency supplied by the pulse generator may change by not more than 300 Hz ***within 20 ms***.

Discrepancy between the Counter Inputs

The discrepancy time for both counters is permanently set to approx. 1 ms. During this period, the S5-95F tolerates different counter statuses in both subunits. If the counter statuses in the two subunits are faulty for longer than 1 ms, the S5-95F responds with a fault signal and with the programmed system reaction.

4.5.1 Connection of Counter Inputs

Example: A pulse generator is to be connected to counter A (IW 36).

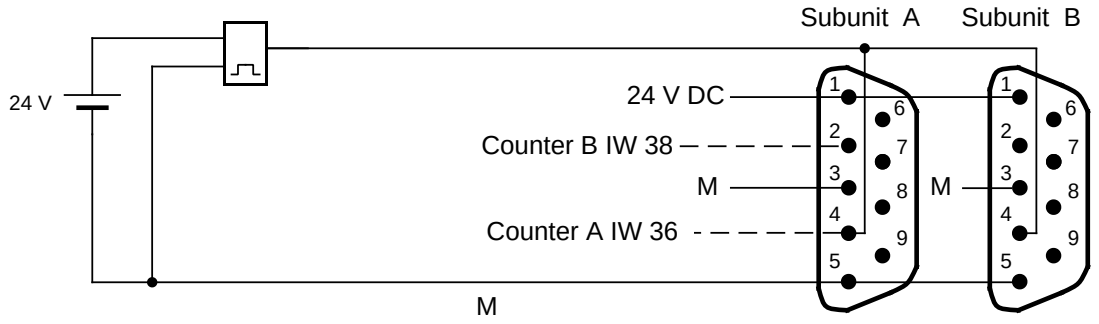


Figure 4-16. Connecting a Counter Input

Counter B (IW 38) is connected in the same way as counter A.

4.5.2 Parameterizing Counter Inputs

In order to utilize the onboard counter inputs function, the counters must first be initialized. To do so, enter the following for each counter input:

- Whether the counter is to be used for counting or for frequency monitoring/rotational speed monitoring
- Whether the counter should count the positive or negative edge of a pulse
- The value up to which the counter should count (comparison value)
- To which signal group the counter is to be assigned.

Make sure when presetting the comparison value that the counter does not trigger OB3 interrupts too frequently. You should therefore select a comparison value which is not too small; the recommended value for the period between two OB3 interrupts is approx. 20 ms.

The following applies irrespective of parameterization:

- The counters only count up
- The maximum counting frequency is 900 Hz (see section 4.5)
- The counter inputs can be used as OB3 interrupt DIs when appropriately shielded.

Parameterization of the Signal Group for Onboard Counters

Note

If you use the onboard counters for safety-related counting tasks, the system reaction for the selected signal group must be "STOP" or "Passivation".
For non-failsafe functions, you may also select standard value generation after ”AND”, ”OR” or ”OLD VALUE” as system response to discrepancies in onboard counters. If there are counter discrepancies, the S5-95F continues processing in this case only with the counter value of subunit A.

4.5.3 Using Counter Inputs for Counting Tasks

The counters must be parameterized in DB1 (see COM 95F manual).

Using Counters A and B Separately

Counters A and B count independent of each other. When the counters reach the parameterized comparison value (counter overflow), then

- an interrupt is triggered and OB3 is called, if it has been programmed
- bit 61.0 in the IB 61 diagnostic byte is set to "1" (for counter A), irrespective of the presence of OB3
- bit 61.1 in the IB 61 diagnostic byte is set to "1" (for counter B), irrespective of the presence of OB3
- the counter is reset to "0", irrespective of the presence of OB3.

The "set" bits in the diagnostic byte, indicating a counter overflow can be reset in OB3 with the following operations: R I 61.0 for counter A or
R I 61.1 for counter B.

Table 4-8. Resetting the Diagnostic Byte in the Program after a Counter Overflow

Example	STL	Explanation
After counter overflow, the corresponding bit in the diagnostic byte is to be reset.	OB3 A I 61.0 R I 61.0 JC FB3 . . . BE	 Counter A counted until comparison value was reached, interrupt was triggered. Reset bit 0 in the diagnostic byte to "0". The counter reaction program in FB3 is executed.

Cascading of Counters A and B

Instead of using counters A and B individually and independently of each other, you can combine (cascade) the two counters. Use cascading counters if you want to count amounts larger than 65,535 (up to 4,294,967,295). The registers of both counters work together like a large counter with more digits. If you cascade both counters, counter input B is automatically disabled; that is why you must transfer the incoming pulses to counter input A.

If the cascaded counter exceeds the comparison value, then

- the interrupt is triggered and OB3 is called, if it has been programmed,
- bit 61.0 and bit 61.1 in the diagnostic byte are set simultaneously to "1", independent of the presence of OB3,
- the counter is reset to "0".

The "set" bits in the diagnostic byte indicating a counter overflow or reaching of the comparison value can be reset with the operations R I 61.0 and R I 61.1.

You can scan the counter status and reset it to "0". The same operations are available as for the individual counters.

4.5.4 Using Counter Inputs for Frequency Monitoring and Monitoring Rotational Speed

Each of the two onboard counter inputs can also be used for frequency monitoring or failsafe monitoring of rotational speed.

To enable the ”frequency monitoring/rotational speed monitoring” function, you must
- initialize the appropriate counter input in DB1 (see COM 95F manual) and
- set a specific bit in output byte QB 35 in the start OBs (OB 21, OB 22).

Set bit

- Q 35.6 in the start OBs for frequency monitoring/rotational speed monitoring via counter input A
- Q 35.7 in the start OBs for frequency monitoring/rotational speed monitoring via counter input B

Frequencies of up to 900 Hz can be measured via the counter inputs. The operating system enters

- the frequency at counter A in IW 36
- the frequency at counter B in IW 38

Frequency Metering

The operating system determines the frequency by acquiring the time interval elapsing for a specific number of pulses (positive edges). The number of pulses counted depends on the specified comparison value; the higher the comparison value, the more pulses the operating system counts (see Table 4.9).

In order to enable the monitored frequency to be updated as often as possible, comparison value and maximum measured frequency should be approximately the same (for example, when the maximum frequency for your application is approximately 120 Hz, the comparison value should be no higher than 149 Hz).

Table 4-9. Comparison Values for Frequency Monitoring

Comparison Value for Frequency Monitoring	Number of Pulses Measured
0 Hz to 49 Hz	1
50 Hz to 99 Hz	2
100 Hz to 149 Hz	3
:	:
850 Hz to 900 Hz	18

The following is applicable for frequency measuring:

$$\text{Frequency (Hz)} = \frac{\text{Number of pulses (n)}}{\text{Time required (in sec.)}}$$

Rotational Speed Monitoring

The S5-95F is also capable of ***failsafe monitoring of an upper rotational speed limit value***. Rotational speed monitoring is based on the exceeding of a specific frequency (overrange), and is not suitable for monitoring of a lower limiting value.

The monitoring of rotational speed requires a pulse generator. Using COM 95F, you must initialize the pulse frequency to be reached at the maximum permissible speed. The following formula applies:

$$\text{Max. pulse frequency (Hz)} = \frac{\text{Max. speed per min.}}{60} \times \text{Pulses generated per revolution}$$

The following occurs if the pulse frequency exceeds the assigned comparison value:

- An interrupt is generated and OB 3 called, if programmed.
- Bit 61.0 in diagnostic byte IB 61 is set to "1" (for counter input A), regardless of whether or not OB3 was programmed.
- Bit 61.1 in diagnostic byte IB 61 is set to "1" (for counter input B), regardless of whether or not OB3 was programmed.

Response times for rotational speed monitoring
The response time depends on the maximum permissible pulse frequency programmed with COM 95F as comparison value for the counter.

Table 4-10. Response Times for Rotational Speed Monitoring

Max. Permissible Pulse Frequency	Response Time
0 Hz (zero-speed monitoring)	15 ms after first pulse
1 Hz to 49 Hz	$< \frac{1}{\text{Frequency in Hz}} + 15$ ms
50 Hz to 99 Hz	< 55 ms
100 Hz to 199 Hz	< 45 ms
200 Hz to 900 Hz	< 40 ms

Safety Note

When you change the comparison value in RUN mode (see section 4.5.5), the S5-95F initializes the frequency/rotational speed monitor. For this reason, it is possible that the response time given above may double.

4.5.5 Scanning and Resetting the Counter Status

Table 4-11 shows the different possibilities of scanning the counters. The table is followed by a program example that shows how to enter a new comparison value.

Table 4-11. Addresses and Counter Access Possibilities

Addresses and Counter Access Possibilities	
Location of counters	Counter A: `IB 36 ... 37` Counter B: `IB 38 ... 39`
Scanning the counter status by reading in the PII (the value read in corresponds to the last signal state read into the PII.) by direct I/O access (the value read in corresponds to the current signal state.)	`L IW 36` `L IW 38` `L PW 36` `L PW 38`
Reset counter status to "0" and input new comparison value.	`L KF x`[1] `T PW 36` `T PW 38`

[1] x = Comparison value, up to which the counter should count (you **must** enter this to enable triggering of an interrupt)

Example: Presetting a new comparison value for counter A.

STL	Explanation
`OB3` `A I 61.0` `R I 61.0` `JC FB 10` `BE`	Counter A counted until a comparison value was reached. An interrupt was triggered. Reset the set bit 61.0 in the diagnostic byte. The program branches to FB10.

STL	Explanation
`FB10` `L KF x` `T PW 36` `BE`	Reaction to "comparison value reached" in the user program. Load the new comparison value x into ACCU 1 and set counter A

Example: How to enter the new comparison value 1,280,520 for the cascading counter.

Parameterize the cascading counter in DB1.
Calculate the hexadecimal number of 1,280,520:

Counter-Register - Cascading Counter

	IW 38 (16)											IW 36 (0)											
bin.:	0	0	0	0	0	0	1	0	0	1	1	1	0	0	0	1	0	1	0	0	0	0	1	0	0	0
hex.:	0			1				3				8			A			0			8			

Use the programmer to enter the following STL into the programmable controller.

STL	Meaning
`FB11` `L KH 0013` `L KH 8A08` `T PW 36` `BE`	Response to ”comparison value reached”. Load comparison value into ACCU1 and ACCU2 and update Transfer statement T PW 36 (the contents of both ACCUs are transferred in T PW 36 only for cascading counter.)

4.6 Failure and Monitoring of the Supply Voltages

The S5-95F monitors the 24 V DC supply voltage for the onboard I/Os. You can evaluate the status of these voltages in the diagnostic byte IB 35 (see section 15.8).

Table 4-12. Monitoring the Supply Voltages for Onboard I/Os

Onboard I/Os	No Supply Voltage for I/Os...	Response from the S5-95F
Onboard DIs/DQs (40-pin connector)	... in one or both subunits	• Error flagged in DB254 (no further system response, such as passivation)
Onboard interrupt DI/ counter inputs (9-pin Sub D connector), but ***no*** counter inputs configured.	... in one subunit	• Error flagged in DB254 (no further system response, such as passivation)
	... in both subunits	• ***No errors*** System assumes that neither counters nor interrupt DIs are being used. If you are using interrupt DIs, you must evaluate IB 35 in the user program.
Onboard interrupt DI/counter inputs and external counter inputs configured.	... in one or both subunits	• OB3 is executed • Error is flagged in DB254 • Programmed system response initiated (for instance, passivation)

Note that I/O errors and the corresponding system response may also be a consequence of a load voltage failure.

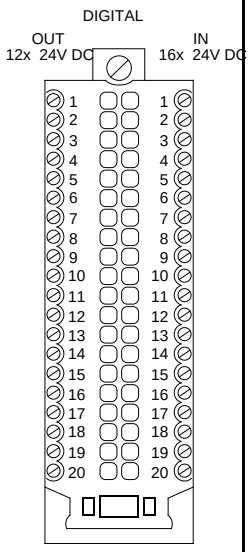
Organizing a Response to a Load Voltage Failure in the User Program

> **Note**
>
> If you want to react to a load voltage failure, you must evaluate diagnostic byte IB 35 in the user program.

4.7 Connector Pin Assignment of the Onboard I/Os

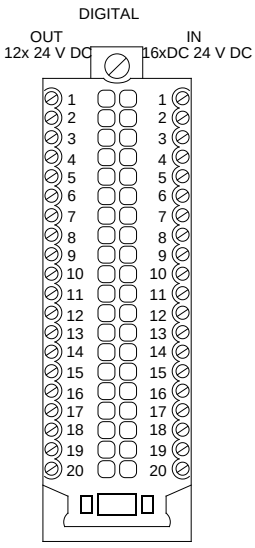
Assignment of the 40-Pin I/O Connector (Left Side of Connector)

Table 4-13. Assignment of the 40-Pin I/O Connector (Left Side of Connector)

Connector	Terminal Number Subunit A	Terminal Number Subunit B	Assignment
	1	1	L+of the 24 V DC load voltage for outputs Q 32.0 to Q 32.7
	2	6	Q 32.0 (P output in subunit A, M output in subunit B)
	3	7	Q 32.1 (P output in subunit A, M output in subunit B)
	4	8	Q 32.2 (P output in subunit A, M output in subunit B)
	5	9	Q 32.3 (P output in subunit A, M output in subunit B)
	6	2	Q 32.4 (M output in subunit A, P output in subunit B)
	7	3	Q 32.5 (M output in subunit A, P output in subunit B)
	8	4	Q 32.6 (M output in subunit A, P output in subunit B)
	9	5	Q 32.7 (M output in subunit A, P output in subunit B)
	10	10	Reference potential (M) for outputs Q 32.0 to Q 32.7
	11		L+ of the 24 V DC load voltage for outputs Q 33.0 to Q 33.3
	12		Q 33.0
	13		Q 33.1
	14		Q 33.2
	15		Q 33.3
	20		Reference potential (M) for outputs Q 33.0 to Q 33.3
		11	L+ of the 24 V DC load voltage for outputs Q 34.0 to 34.3
		12	Q 34.0
		13	Q 34.1
		14	Q 34.2
		15	Q 34.3
		20	Reference potential (M) for outputs Q 34.0 to Q 34.3

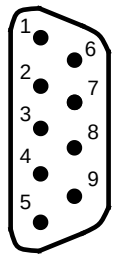
Assignment of the 40-Pin I/O Connector (Right Side of Connector)

Table 4-14. Assignment of the 40-Pin I/O Connector (Right Side of Connector)

Connector	Terminal Number Subunit A and B		Assignment
	1	1	L+ of the 24 V DC load voltage for inputs I 32.0 to I 33.7
	2	2	I 32.0
	3	3	I 32.1
	4	4	I 32.2
	5	5	I 32.3
	6	6	I 32.4
	7	7	I 32.5
	8	8	I 32.6
	9	9	I 32.7
	10	10	
	11	11	
	12	12	I 33.0
	13	13	I 33.1
	14	14	I 33.2
	15	15	I 33.3
	16	16	I 33.4
	17	17	I 33.5
	18	18	I 33.6
	19	19	I 33.7
	20	20	Reference potential (M) for inputs I 32.0 to I 33.7

Assignment of the 9-Pin Sub D Connector

Table 4-15. Assignment of the 9-Pin Sub D Connector

Terminal	Terminal Number Subunit A and B		Assignment
	1	1	L+ of the 24 V DC voltage
	2	2	Input of counter B (IW 38)
	3	3	Reference potential (M) for counters and interrupt inputs
	4	4	Input of counter A (IW 36)
	5	5	Reference potential (M) for counters and interrupt inputs
	6	6	Interrupt input I 59.0
	7	7	Interrupt input I 59.1
	8	8	Interrupt input I 59.2
	9	9	Interrupt input I 59.3

Note

To ensure proper functioning of the S5-95F you must screw the sub D connector to the basic unit.

5 Expansion of Basic System with External I/Os

5.1	Assembling a Tier	5 - 1
5.2	Multi-Tier Expansion	5 - 5
5.3	Cabinet Mounting	5 - 6
5.3.1	Horizontal Mounting	5 - 7
5.3.2	Vertical Mounting	5 - 8
5.4	Connection Methods: Screw-Type Terminals and Crimp Snap-In	5 - 9
5.5	Connection of Failsafe I/O Modules	5 - 12
5.5.1	Expansion of Basic Unit with Failsafe Digital Input Module	5 - 13
5.5.2	Expansion of Basic Unit with Failsafe Digital Output Module	5 - 15
5.5.3	Expansion of Basic Unit with Failsafe Analog Value Processing	5 - 21
5.6	Connection of Non-Failsafe I/O Modules	5 - 23
5.7	Power Supplies for the S5-95F	5 - 27
5.8	Electrical Potentials for the Onboard and External I/Os	5 - 28
5.8.1	Onboard I/Os of the S5-95F Programmable Controller	5 - 28
5.8.2	Potential Bonding and Galvanic Isolation of External I/Os	5 - 29
5.9	Grounded or Non-Grounded S5-95F Configuration	5 - 31

Figures

5-1 Installation of a Basic Unit with External I/Os . . . . . 5 - 1
5-2 Connecting the Bus Units . . . . . 5 - 2
5-3 Coding System to Prevent an Inadvertent Interchange of Modules . . . . . 5 - 3
5-4 Covering the Last Slot with Connector Cover 981-8MA11 . . . . . 5 - 4
5-5 Interconnecting Tiers with Interface Modules (6ES5 316-8MA12) . . . . . 5 - 5
5-6 Multi-Tier Configuration of a Subunit with the IM 316 Interface Module (6ES5 316-8MA12) . . . . . 5 - 7
5-7 Cabinet Mounting with a Series of Devices and/or Cable Duct . . . . . 5 - 7
5-8 Vertical Mounting of a PLC . . . . . 5 - 8
5-9 Screw-Type Terminal for Power Supply Modules and Bus Units . . . . . 5 - 9
5-10 Installing a Crimp Snap-In Terminal . . . . . 5 - 10
5-11 Disconnecting a Terminal . . . . . 5 - 11
5-12 Example for Connection of Single-Channel, Failsafe Sensors to an External Digital Input Module . . . . . 5 - 13
5-13 Example for Connection of Two-Channel Sensors to an External Digital Input Module . . . . . 5 - 14
5-14 Example for Connection of the External Digital Output Module with Direct P-M Control of the Actuator . . . . . 5 - 17
5-15 Example for Indirect Control of Actuators (one P output and one M output) . . . . . 5 - 18
5-16 Example for Connection of the External Digital Output Module (twice P output) . . . . . 5 - 19
5-17 Example of a Connection for Single-Pole, Two-Channel Actuator Control . . . . . 5 - 20
5-18 Example for Connection of Two Current Sensors . . . . . 5 - 22
5-19 Connecting a Sensor to Channel 4 . . . . . 5 - 23
5-20 Connecting a Lamp to Channel 6 . . . . . 5 - 24
5-21 Front View of the Digital I/O Module with a Crimp Snap-In Connector (simplified view and not true to scale) . . . . . 5 - 25
5-22 Connecting a Sensor and a Lamp to Digital Input/Output Module 482 . . . . . 5 - 26
5-23 Simplified Representation of Electrical Potentials for the Basic Unit . . . . . 5 - 28
5-24 Simplified Representation of a Non-Floating, External I/O Connection . . . . . 5 - 29
5-25 Simplified Representation of a Floating, External I/O Connection . . . . . 5 - 30
5-26 Grounded S5-95F Configuration . . . . . 5 - 31
5-27 Non-Grounded S5-95F Configuration . . . . . 5 - 32

Tables

5-1 Pin Assignment for 450-8FAxx Digital Output Module . . . . . 5 - 15
5-2 Blanking Times for DQ 450-8FAxx . . . . . 5 - 15

5 Expansion of Basic System with External I/Os

You can expand the programmable controllers by using S5-100U external I/O modules. All modules certified as reaction-free (isolated) are permitted (see Appendix A).

If you install S5-100U external I/O modules, mount the basic unit, bus units, and interface module on a 35-mm standard mounting rail and connect them to each other via the ribbon cables of the bus units. Each bus unit has room available for two I/O modules.

Figure 5-1 shows a basic unit expanded with external I/Os.

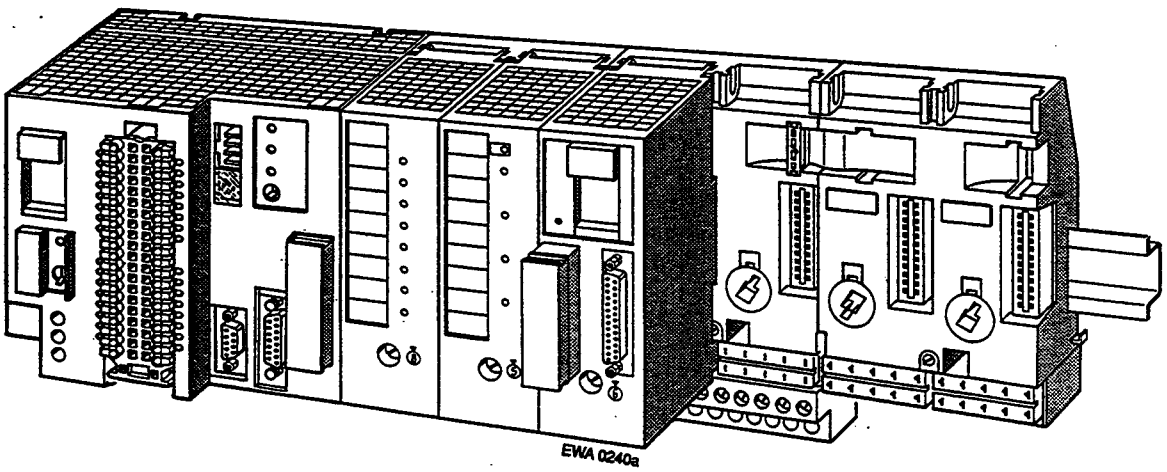


Figure 5-1. Installation of a Basic Unit with External I/Os

5.1 Assembling a Tier

To expand a basic system with external I/O modules, you need the following parts for each subunit:

- An S5-95F basic unit
- Bus units (max. 8)
- I/O modules
- Standard mounting rail

Mount the first module on the far left end of the standard mounting rail and then add the other modules to the right of the first module.

How to Mount and Remove the Bus Units

Bus units are also mounted on a standard mounting rail. Mount the bus units in the same way you mounted the controller or a power supply module.

There are small hooks on the sides of the bus units that interlock them.

- Hook the bus unit onto the top of the standard mounting rail beside the programmable controller.
- Press the bus unit down firmly until the slide audibly snaps into place.

To remove bus units, proceed as follows:

- Loosen the connections (flat ribbon cable) to the adjoining devices.
- Using a screwdriver, press down on the slide.
- Swing the bus unit up and out of the standard mounting rail.

How to Connect Bus Units to the Basic Unit or Interconnecting Bus Units

- Pull the flat ribbon cable connector (top left on the bus unit) out of its holder.
- Plug the connector into the receptacle on the right side of the basic unit or into the adjacent receptacle of the bus unit on the left side (see Figure 5-2).

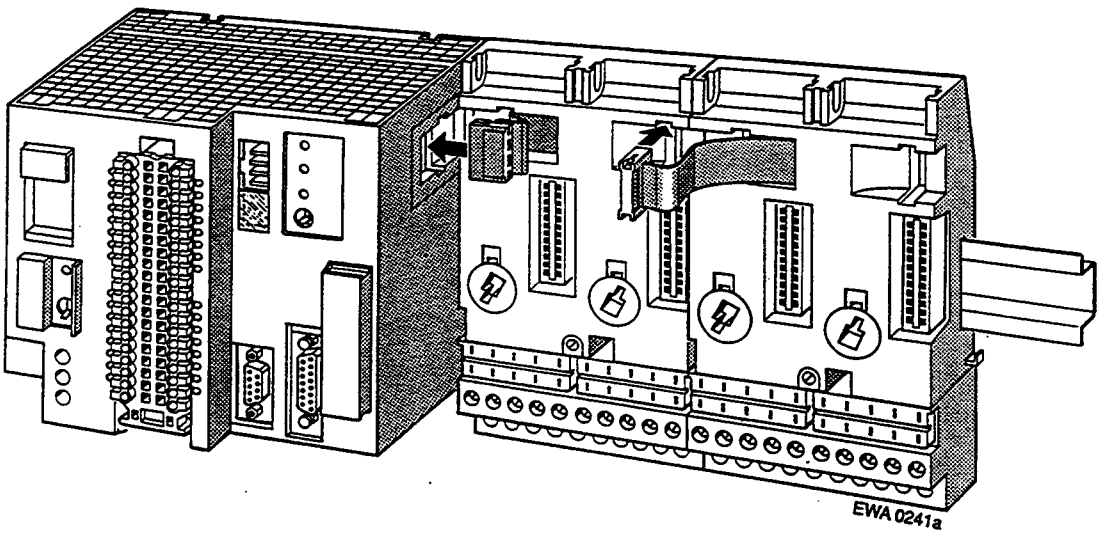


Figure 5-2. Connecting the Bus Units

How to Plug Input and Output (I/O) Modules into the Bus Units

Before you mount an input or output module, you must set the coding element on the bus unit to match the module type. The coding element keeps you from confusing module types when exchanging modules.

Use the following information to set the coding element.

A code number is printed on the frontplate of every I/O module. The number is between two and eight, depending on the particular module type. There is a white, mechanical coding key located on the back of each module. The position of the coding key is determined by the module type and cannot be changed. The bus unit has a mating component for each key, a white rotating coding element or "lock" (see Figure 5-3).

- Use a screwdriver to set the "lock" on the bus unit to the corresponding I/O module code number.

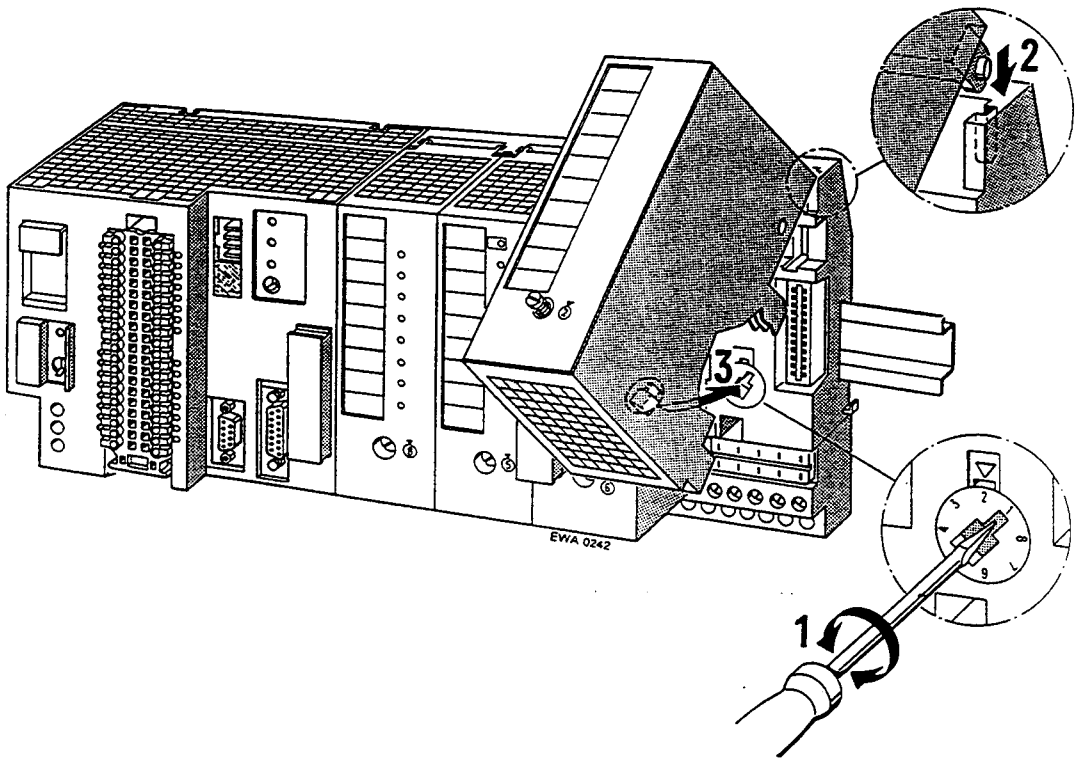


Figure 5-3. Coding System to Prevent an Inadvertent Interchange of Modules

To attach the module, proceed as follows:

- Hook the module onto the top of the bus unit.
- Swing the module down onto the bus unit.
- Press the module down firmly.
- Tighten the hold-down screw on the front of the module to attach the module to the bus unit

To detach I/O modules:

- Loosen the hold-down screw and swing the module up and out of the bus unit.

Note on the Use of Bus Units in the S5-95F

If the ***last slot*** in a tier is ***not occupied***, you must fit it with a connector cover as a precaution against electrostatic discharge (see Fig. 5-4). Connector cover 981-8MA11 is required.

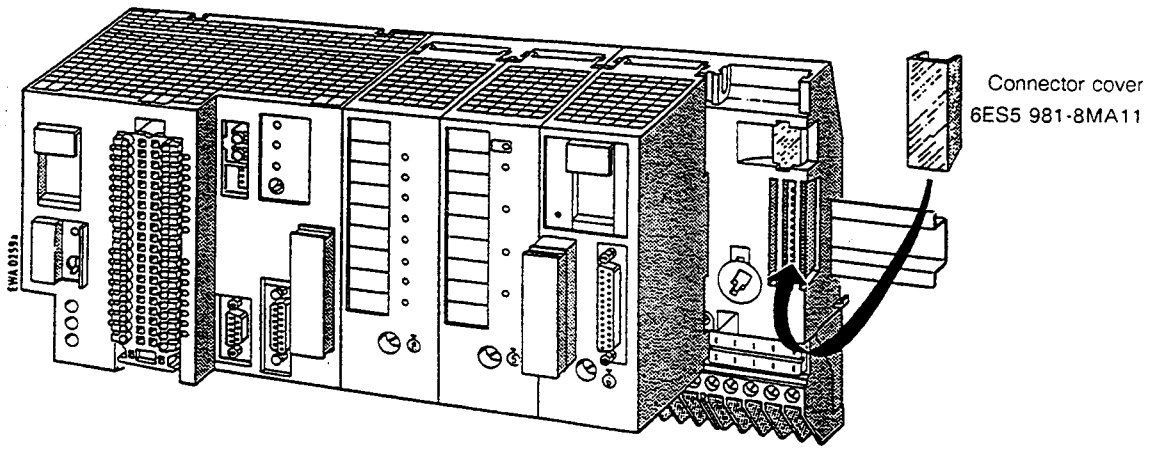


Figure 5-4. Covering the Last Slot with Connector Cover 981-8MA11

5.2 Multi-Tier Expansion

If it is not possible to have all of the modules located on one tier, then you can expand the configuration up to four tiers. You may use a maximum of 8 bus units for each subunit. It does not matter how many bus units are mounted on a tier. You need one interface module per tier to connect the tiers one to the other.

Install an interface module as you would install a bus unit. You must connect each interface module to the last bus unit via the flat ribbon cable.

Use the IM 316-8FA12 interface modules for multi-tier configurations. Use the 712-8 connecting cable to connect the IM 316 interface modules (Order No. 6ES5 712-8...).

The standard mounting rails must have a common reference potential if they are mounted in different cabinets.

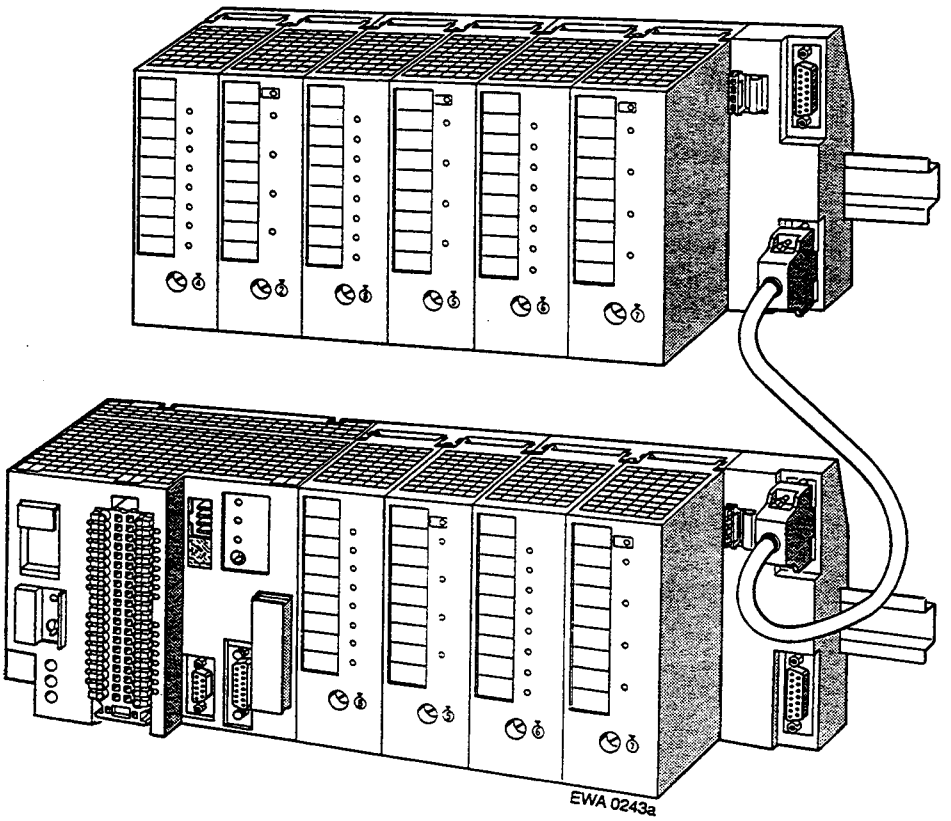


Figure 5-5. Interconnecting Tiers with Interface Modules (6ES5 316-8MA12)

Safety Note

For reasons of EMC you must cover unused sub D locations on the IM 316 interface module with an empty sub D connector (6ES5-750-2AA21).

How to Install the Interface Module

Hook the interface module to the standard mounting rail.
Swing the interface module back until the slide on the bottom snaps into place on the rail.
Use the flat ribbon cable to connect the module to the last bus unit.
Use connecting cable 712-8 to join the two interface modules.
Connect the cable to the "out" socket on the PLC tier and to the "in" socket on the expansion tier.
Securely screw the connecting cable plugs in place.
Plug the caps onto the unused sub D connectors.

How to Remove the Interface Module

Only for IM 316: Loosen the hold-down screws from the plugs and remove the connecting cable.
Remove the connecting flat ribbon cable from the adjacent bus unit.
Use a screwdriver to press down on the slide located on the bottom of the interface module.
Swing the module up and out of the standard mounting rail.

5.3 Cabinet Mounting

To improve noise immunity, the programmable controller should be mounted on a metal plate. There should be electrical continuity between the grounded enclosure and the mounting rails. Make sure that the system is bonded to earth.

You can use the 8LW system or the 8LX system mounting plates (see Catalog NV21).

Adequate ventilation and heat dissipation are important to the proper operation of the system. You must have at least 210 mm (8.3 in.) between each mounting rail (see Figures in Appendix B) for proper ventilation.

Locate the programmable controller and the power supply on the lowest tier.

To provide adequate cabinet ventilation, define the total heat loss by calculating the sum of all typical heat losses (see Catalog ST 52.1).

5.3.1 Horizontal Mounting

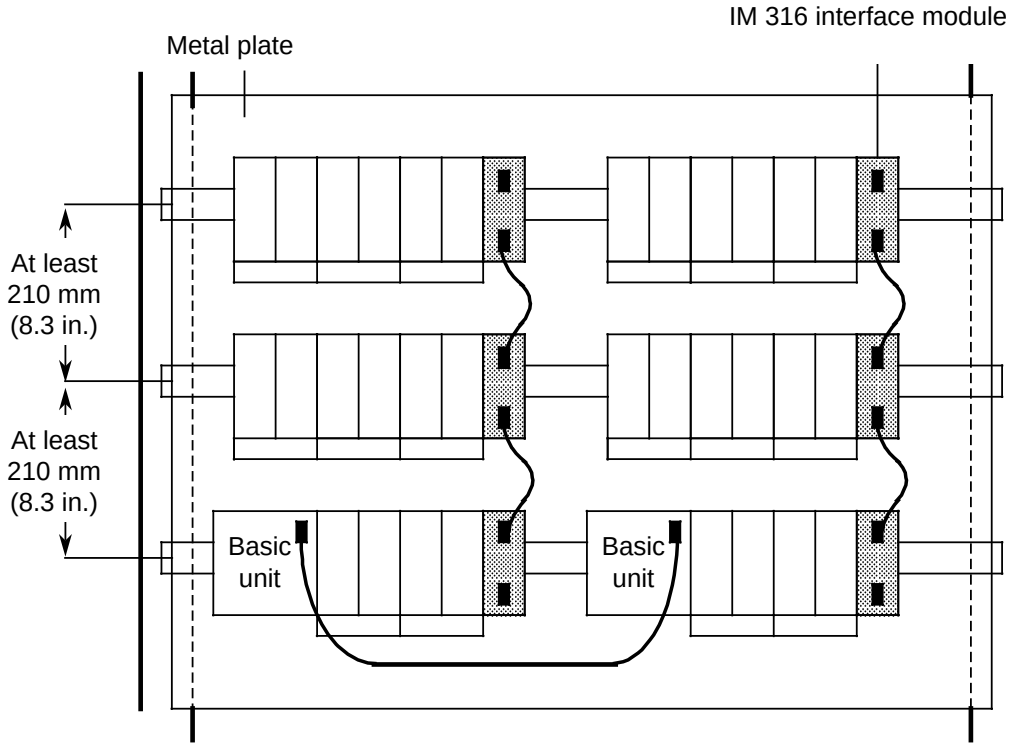


Figure 5-6. Multi-Tier Configuration of a Subunit with the IM 316 Interface Module (6ES5 316-8MA12)

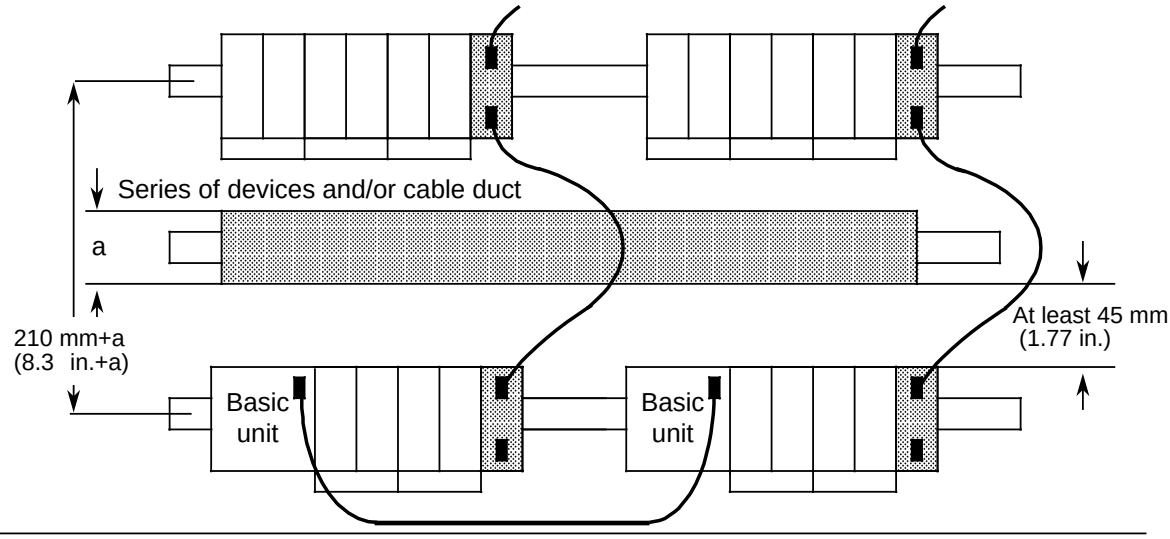


Figure 5-7. Cabinet Mounting with a Series of Devices and/or Cable Duct

5.3.2 Vertical Mounting

You can also mount the standard mounting rails vertically and then attach the modules one over the other. Because heat dissipation by convection is less effective in this case, the maximum ambient temperature allowed is 40° C (104° F).

Use the same minimum clearances for a vertical configuration as for a horizontal configuration.

You must install a clamp (see Catalog SA 2) on the lower end of the PLC tier to hold the modules in position.

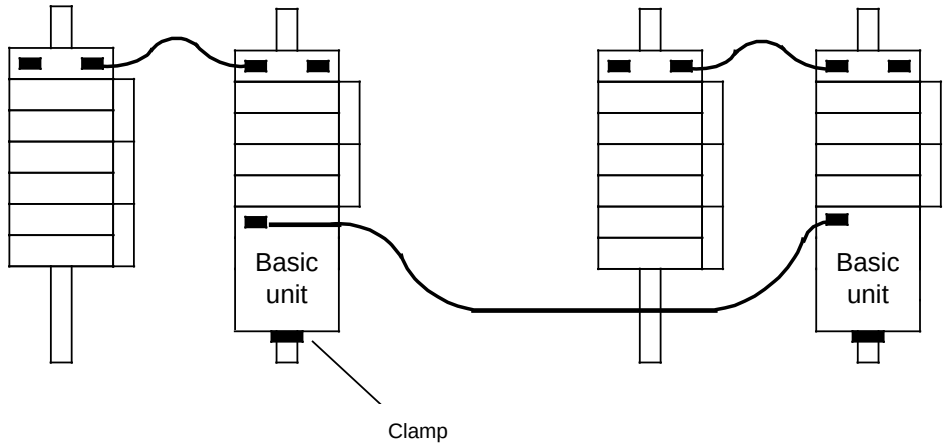


Figure 5-8. Vertical Mounting of a PLC

5.4 Connection Methods: Screw-Type Terminals and Crimp Snap-In

Screw-Type Terminal (for the 40-Pin Connector)

When using the screw-type terminals of the 40-pin connector, you can connect cables that do not exceed the maximum cross-section of 1.5 mm^2. It is best to use a 3.5-mm screwdriver to tighten the screws.

Permissible cable cross-sections are:

- A solid conductor: 1 x 0.25 to 1.5 mm^2
- A stranded conductor with a core end sleeve: 1 x 0.25 to 1.5 mm^2
  Use only 10 mm long core end sleeves to DIN 46228 for reasons of EMC.

Screw-Type Terminal for Power Supply Modules and Bus Units

You can wire two cables per terminal using this screw-type terminal. It is best to tighten the screws using a 5-mm screwdriver.

Permissible cable cross-sections are:

- A solid conductor: 2×0.5 ... 2.5 mm^2
- A stranded conductor with a core end sleeve: 2×0.5 ... 1.5 mm^2
  Use only 10 mm long core end sleeves to DIN 46228 for reasons of EMC.

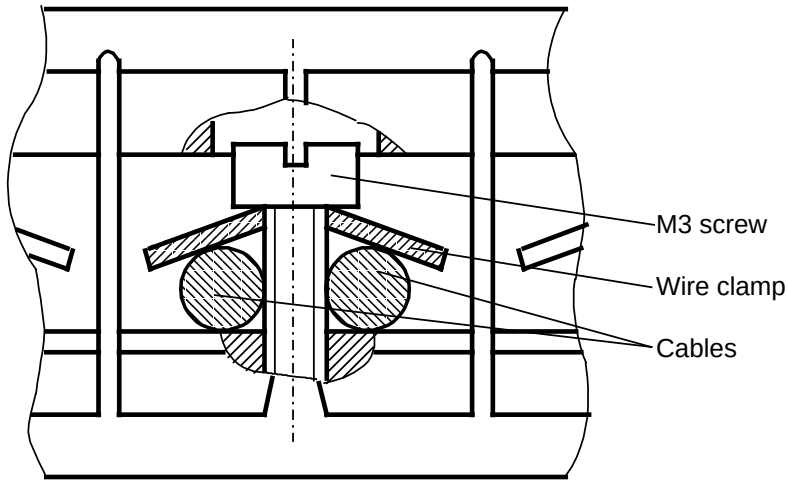


Figure 5-9. Screw-Type Terminal for Power Supply Modules and Bus Units

Crimp Snap-In Terminals

Bus units using the crimp snap-in connection method have exactly the same height as the S5-95F basic unit. You can connect stranded conductors with a cross-section of between 0.5 to 1.5 mm^2 to these terminals.

How to Connect the Contact to the Terminal Block (see Figure 5-10)

- Remove the module that is plugged into the bus unit.
- Use a screwdriver to press down on the terminal block (1).
- Swing the terminal block up. The rear side is now visible (2).
- Push the contact into the desired opening until the locating spring engages.
  Caution: The spring must point into the slot!
- Pull lightly on the cable to make certain that the contact is properly engaged.
- Swing the terminal block back into its original position. Press up on the terminal block until it snaps into position.

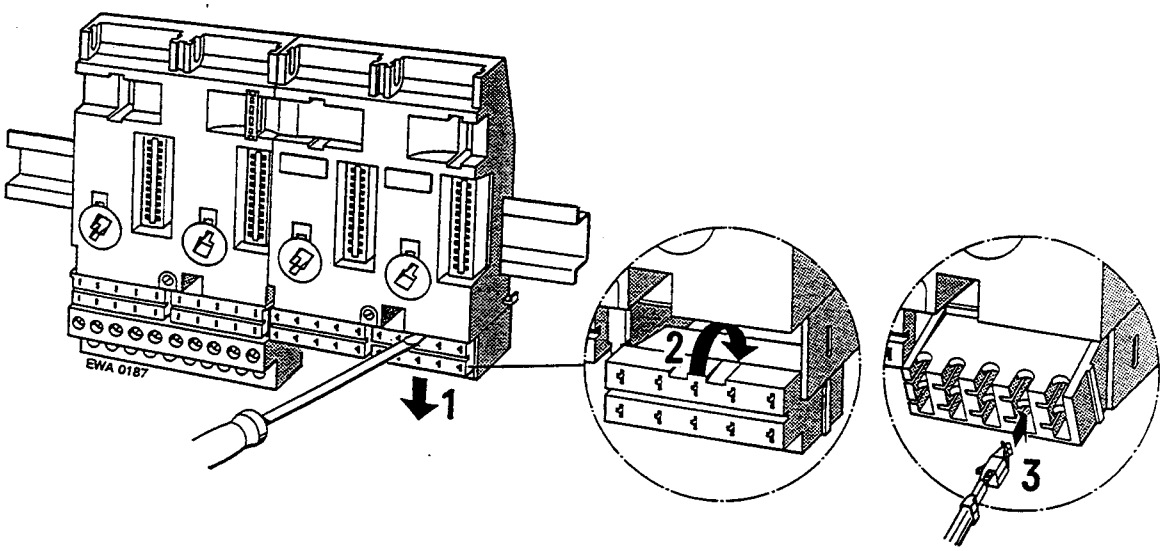


Figure 5-10. Installing a Crimp Snap-In Terminal

How to Disconnect the Terminal Block

- Position the terminal block as is shown in Figure 5-11.
- Insert the extraction tool into the slot beside the terminal so that you can compress the barb.
- Position the cable in the groove on the extraction tool and pull out both the tool and the cable.
- Realign the deformed barb so that you can use the terminal again.

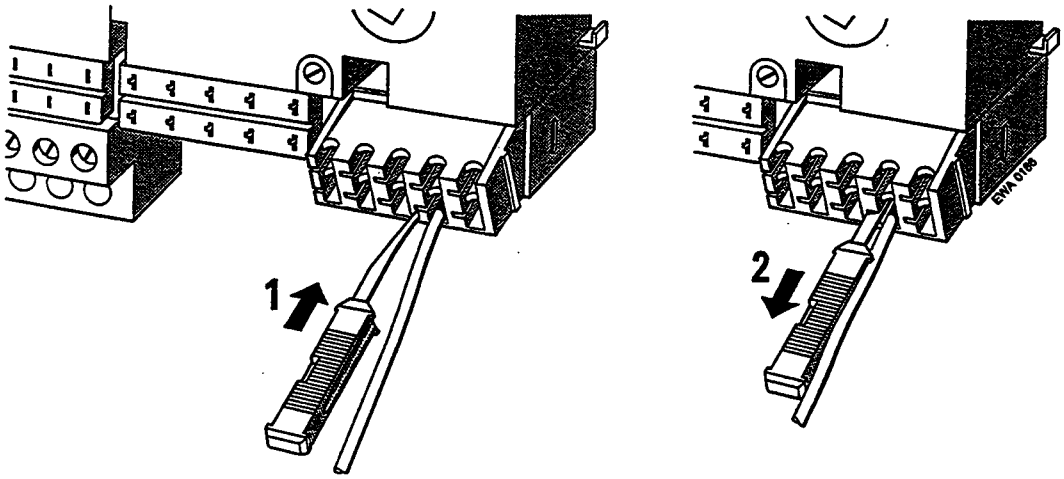


Figure 5-11. Disconnecting a Terminal

5.5 Connection of Failsafe I/O Modules

If the failsafe onboard inputs and outputs are not sufficient for your application, you can expand the S5-95F with the following I/O modules:

- Failsafe input module DI 431-8FA..
- Failsafe output module DQ 450-8FA..

You can use the I/O modules DI 431-8FA.. and DQ 450-8FA.. both in single-channel and two-channel operation. For safety functions, you must always use the modules in two-channel operation and parameterize them with COM 95F.

No matter whether you operate the above-mentioned modules in single or two-channel operation, the modules are subject to a high-grade function test.

> **Note**
>
> Make sure that the load voltage for all failsafe external I/O modules is connected. If the load voltage is not connected, the S5-95F reacts with a fault message and with the safety reaction programmed in DB1.

System Reaction to I/O Faults

The reaction to an I/O fault depends on whether you use the I/O modules in single or two-channel operation.

When you use the I/O modules in ***two-channel*** operation, the S5-95F reacts as follows:

- The yellow fault LED is set
- The fault is entered into the system event DB 254
- Error OB 37 is processed
- System reaction as programmed in DB1

When you use the I/O modules in ***single-channel*** operation, the S5-95F reacts as follows:

- The yellow fault LED is set
- The fault is entered into the system event DB 254
- Error OB 37 is processed
- The user program is executed (no further system reaction)

5.5.1 Expansion of Basic Unit with Failsafe Digital Input Module

When you use the digital input module 431-8FA.. for safety-related tasks, you must always use it in a redundant configuration (in pairs) - one module in subunit A and one module in subunit B. Eight galvanically isolated DI channels are available for each pair of modules.

Example: Connection of Single-Channel, Failsafe Sensors (Type B)

Insert one module into slot n of subunit A and the other module into slot n+1 of subunit B, with n being an even number between 0 and 30 (see section 6.2).
Connect the 24 V DC supply voltage to terminals 1 and 2.
Wire the inputs with the sensors (as shown in Figure 5-12).
Parameterize the inputs in DB1.

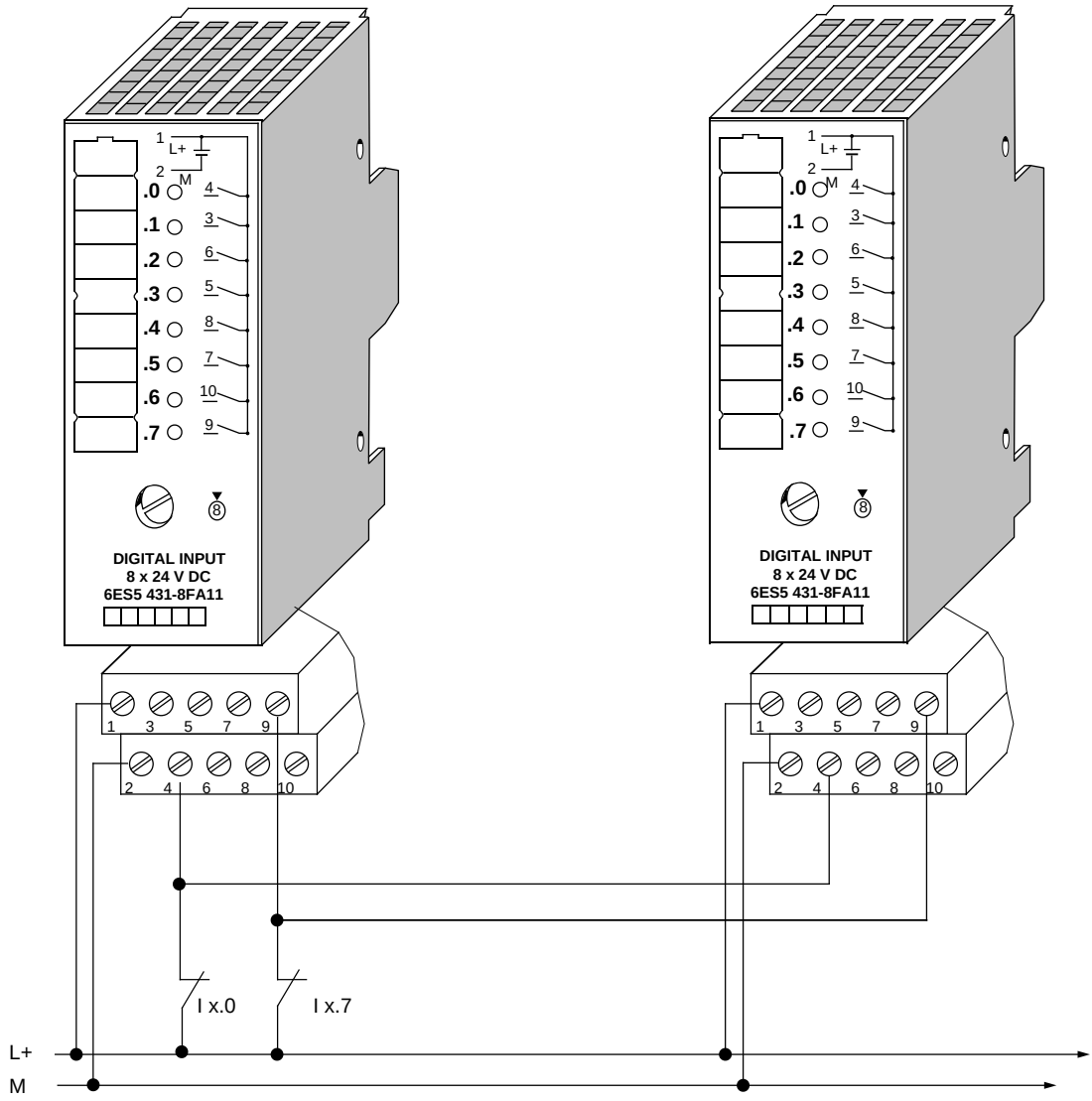


Figure 5-12. Example for Connection of Single-Channel, Failsafe Sensors to an External Digital Input Module

Example: Connection of Two-Channel Sensors (Type C)

Insert one module into slot n of subunit A and the other module into slot n+1 of subunit B, with n being an even number between 0 and 30 (see section 6.2).
Connect the 24 V DC supply voltage to terminals 1 and 2.
Wire the inputs with the sensors (as shown in Figure 5-13).
Parameterize the inputs in DB1.

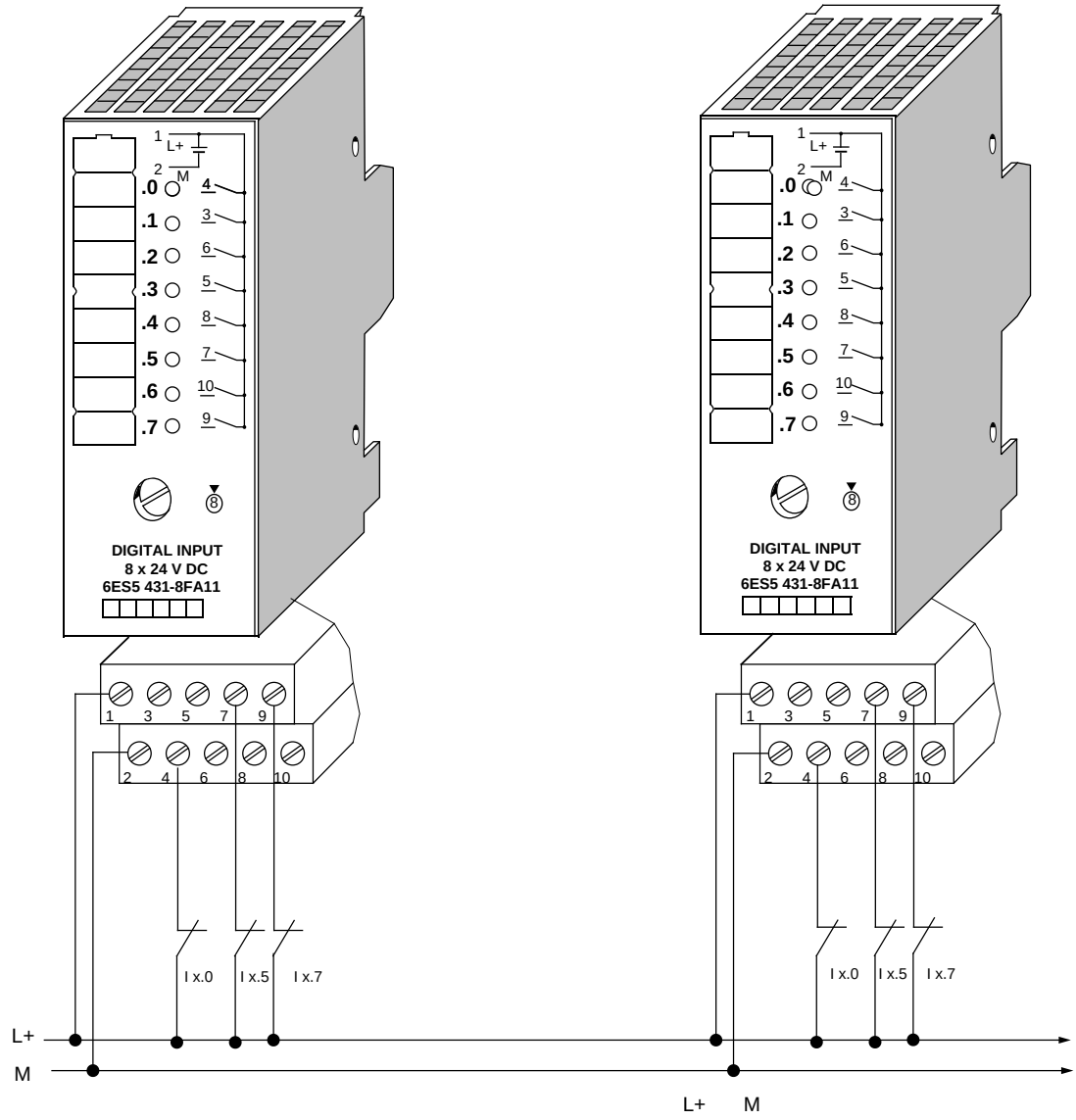


Figure 5-13. Example for Connection of Two-Channel Sensors to an External Digital Input Module

5.5.2 Expansion of Basic Unit with Failsafe Digital Output Module

When you use the digital output module 450-8FAxx for safety-related tasks, you must always use it in a redundant configuration (in pairs) - one module in subunit A and one module in subunit B. Four galvanically isolated DQ channels are available for each pair of modules.

Digital output module 450-8FAxx has 4 digital outputs, which can output either P or M potential. You select the type of potential to be output with a switch on the front of the module.

Please note that the pin assignment for the 450-8FAxx digital output module depends on the switch position selected (P or M potential).

The following table shows the pin assignment of the 450-8FAxx for each switch position.

Table 5-1. Pin Assignment for 450-8FAxx Digital Output Module

Switch Position	24 V DC Power Supply		Output Assignment			
Switch set to	L+ connection	M connection	Bit 0	Bit 1	Bit 2	Bit 3
P	Terminal 1, 3, 5, 7, 9 (jumper)	Terminal 2	Terminal 4	Terminal 6	Terminal 8	Terminal 10
M	Terminal 1	Terminal 2, 4, 6, 8, 10 (jumper)	Terminal 3	Terminal 5	Terminal 7	Terminal 9

Compatibility between Modules DQ 450-8FA11 and DQ 450-8FA12

With the exception of the (shorter) blanking time, module DQ 450-8FA12 is functionally fully upward-compatible to module DQ 450-8FA11. For this reason, it is permissible to drive an actuator via a module pair consisting of one DQ 450-8FA11 and one DQ 450-8FA12. Note, however, that in this case the driven actuator must be oriented to the DQ 450-8FA11 module's longer blanking time.

Table 5.2. Blanking Times for DQ 450-8FAxx

Module	Blanking Time for Module Test
DQ 450-8FA11	< 7 ms
DQ 450-8FA12	< 1 ms

EMC Strength of the DQ 450-8FA12 in the Event of a Surge Pulse to IEC 801-5

If the DQ 450-8FA12 malfunctions due to a surge pulse as defined by IEC 801-5, the S5-95F continues to operate normally. However, the malfunction caused by the surge pulse can trigger the error response programmed for the respective signal group. Even in this case, the S5-95F system remains active at a minimum of operational quality to EN 50082, continuing to execute the necessary safety tests and issue the relevant fault/error messages. The minimum operational quality is dependent on the programmed response:

- If ***STOP*** was programmed as response for the signal group to which the DQ 450-8FA12 is allocated, the S5-95F goes to STOP. The system returns to the operating system level. The self-tests for CPU and communication between the subunits (but not the I/O test) will continue to be executed.
  For reasons of safety, all DQ 450-8FA12s as well as the onboard I/Os shut down their load voltage via an internal relay. The system can be switched back on again by setting it to RUN, then to STOP, then back to RUN, or by a POWER OFF followed by a POWER ON.
- If ***passivation*** was programmed as response for the signal group to which the DQ 450-8FA12 was allocated, all I/Os assigned to this signal group are shut down (all outputs go to "0") while all other I/Os remain active.
  For reasons of safety, the DQ 450-8FA12 shuts down the load voltage via an internal relay. You can return the passivated I/Os to the process in RUN mode via function block FB255 (see section 9.2.10).
- If ***AND/OR*** or ***old value*** was programmed as response for the signal group to which the DQ 450-8FA12 was allocated, the S5-95F responds with the appropriate standard signals.
  If a safety response is required, you must organize it yourself at the user level, for example by evaluating system event DB 254.

Exception for Single-Pole, Two-Channel Actuator Control with DQ 450-8FA12

In addition to the familiar circuit variations for the DQ 450-8FA11, you can implement single-pole, two-channel actuator control with the new DQ 450-8FA12 module. The advantage of this variation is its simpler wiring (see Figure 5-17).

> **Safety Note**
>
> Single-pole, two-channel actuator control (see Figure 5-17) can cause a short circuit in L + which could result in permanent or constant triggering of the actuator. The conductor arrangement must therefore conform to DIN EN 60204 for reasons of safety.

Wiring Inductances Controlled by the DQ 450-8FA11

> **Note**
>
> The induced voltage which occurs when disconnecting inductances can cause an I/O fault to be detected in a digital output module which is, in fact, non faulted. The S5-95F reacts by issuing an error message and performing the parameterized system response (STOP or passivation of the signal group).
> You should therefore wire up inductances controlled via the 450-8FAxx digital output module with a free-wheeling diode (see Figs. 5-14 to 5-17).

Example: Direct Control of Actuators (Type K)

Insert one module into slot n of subunit A and the other module into slot n+1 of subunit B, with n being an even number between 0 and 30 (see section 6.2).
Set the selector switch of the module in subunit A to P (P output) and the selector switch of the module in subunit B to M (M output) or vice versa.
Connect the 24 V DC supply voltage to terminals 1 and 2 (use jumper, see Table 5.1).
Wire the outputs with the actuators (as shown in Figure 5-14).
Parameterize the outputs in DB1.

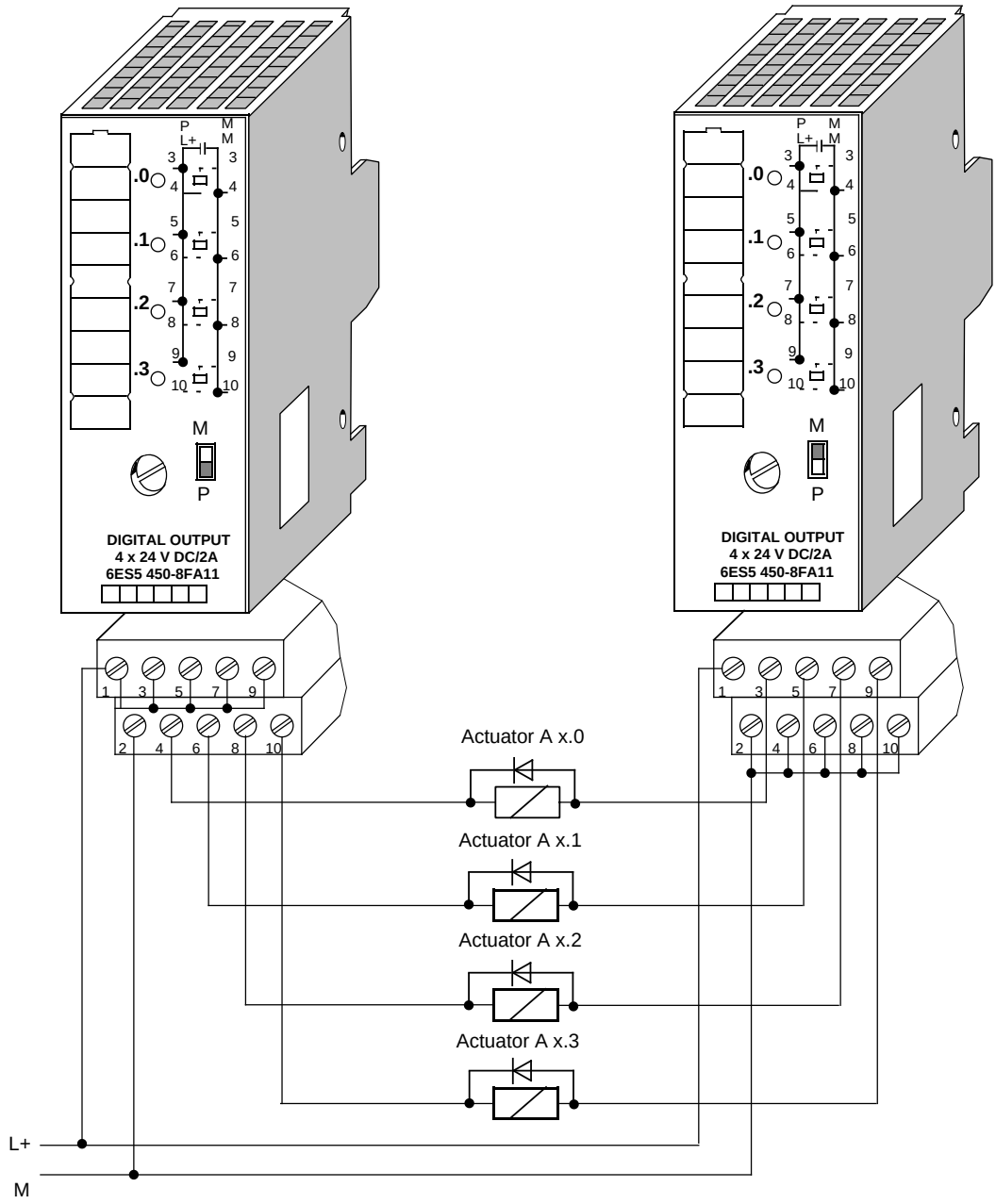


Figure 5-14. Example for Connection of the External Digital Output Module with Direct P-M Control of the Actuator

Example: Indirect Control of Actuators via Coupling Relays with P-M Control (Type L)

Insert one module into slot n of subunit A and the other module into slot n+1 of subunit B, with n being an even number between 0 and 30 (see section 6.2).
Set the selector switch of the module in subunit A to P (P output) and the selector switch of the module in subunit B to M (M output) or vice versa.
Connect the 24 V DC supply voltage to terminals 1 and 2 (use jumper, see Table 5.1).
Wire the outputs with the coupling relays (as shown in Figure 5-15).
Wire the actuators.
Parameterize the outputs in DB1.

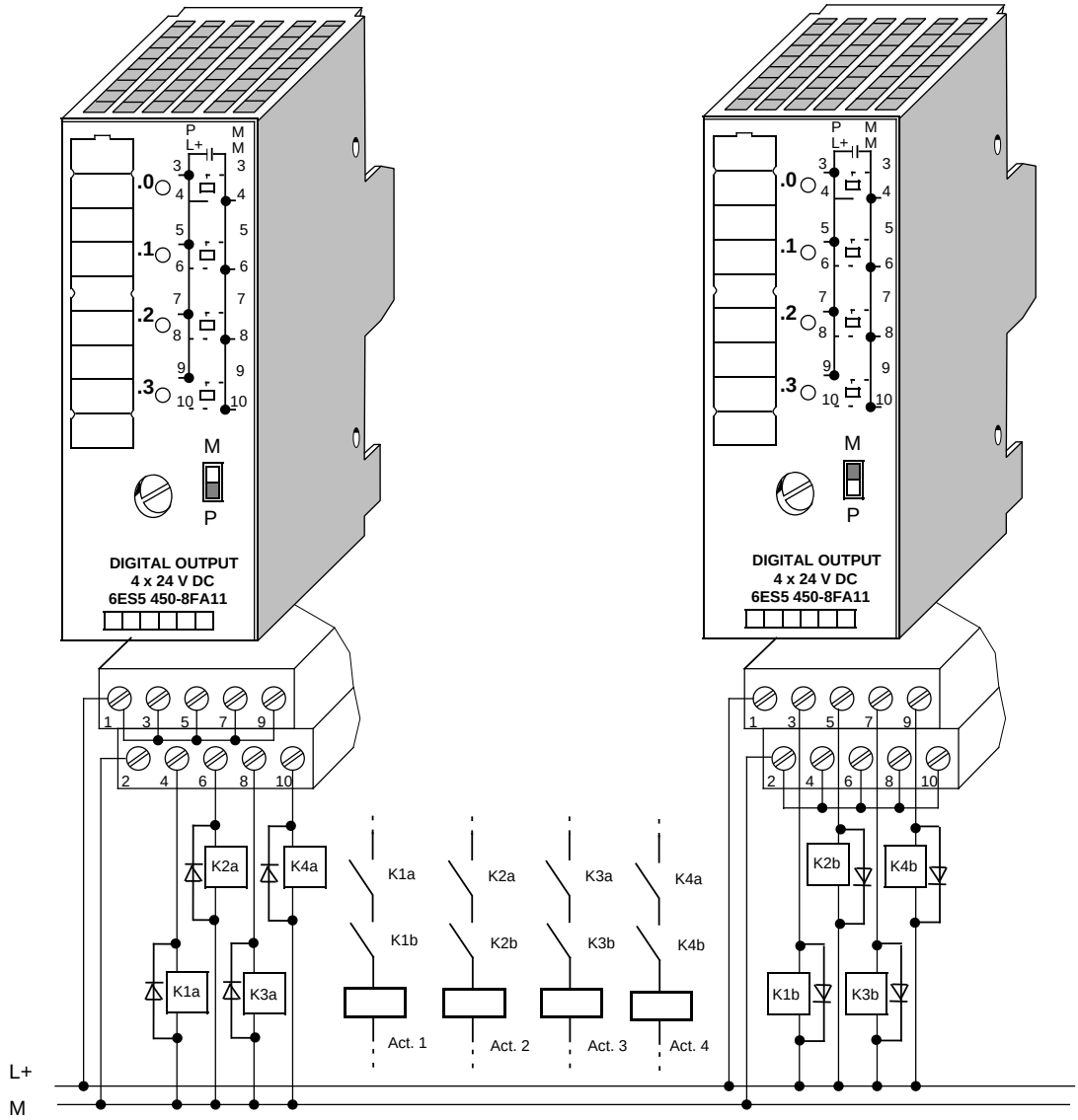


Figure 5-15. Example for Indirect Control of Actuators (one P output and one M output)

An overview of suitable coupling elements is given in section 4.3.

Example: Indirect Control of Actuators via Coupling Relays with P-P Control (Type M)

Insert one module into slot n of subunit A and the other module into slot n+1 of subunit B, with n being an even number between 0 and 30 (see section 6.2).
Set the selector switches of the modules in subunit A and B to P (P output).
Connect the 24 V DC supply voltage to terminals 1 and 2 (use jumper, see Table 5.1).
Wire the outputs with the coupling relays (as shown in Figure 5-16).
Wire the actuators.
Parameterize the outputs in DB1.

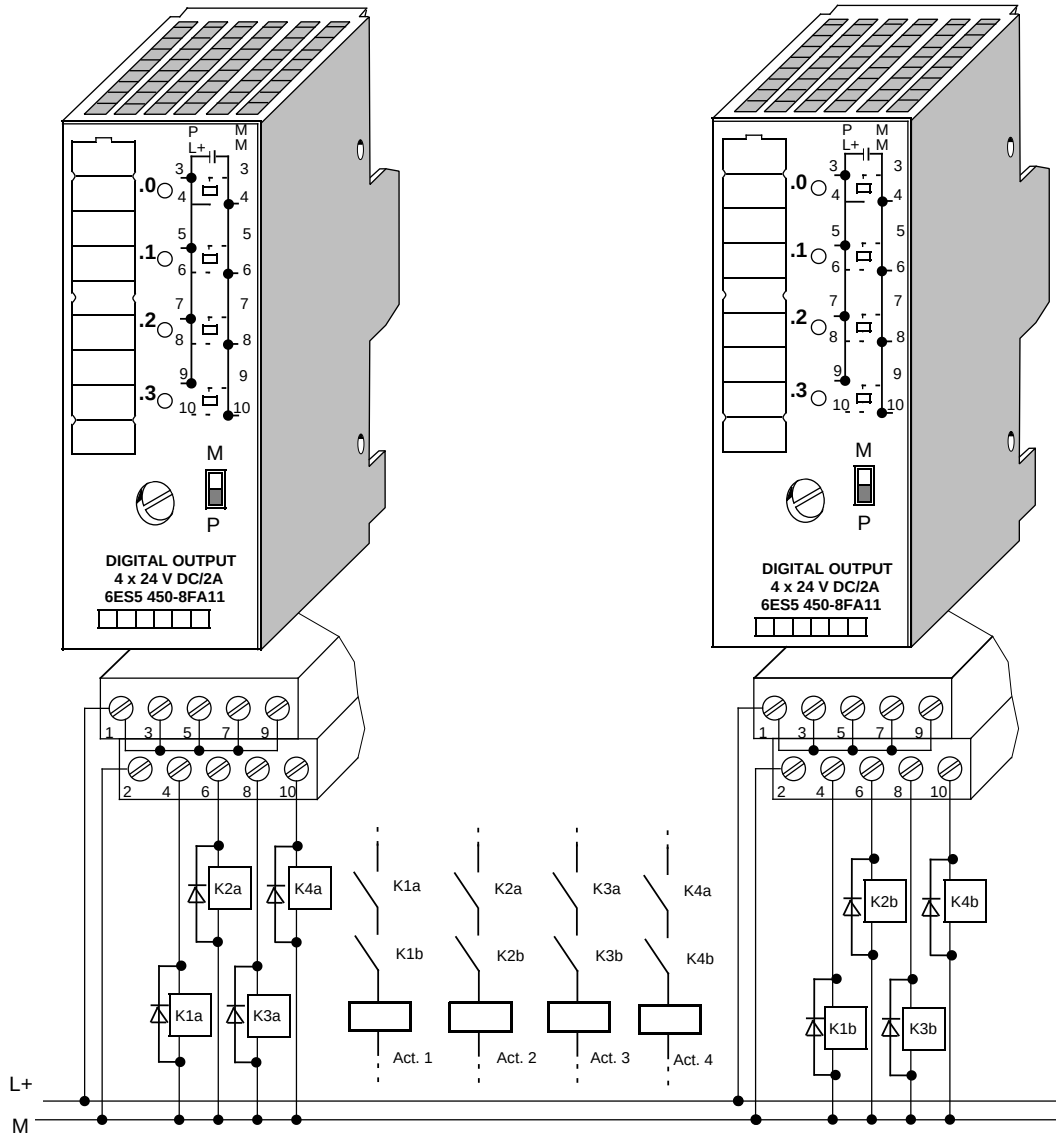


Figure 5-16. Example for Connection of the External Digital Output Module (twice P output)

An overview of suitable coupling elements is given in section 4.3.

Example: Single-Pole, Two-Channel Actuator Control with DQ 450-8FA12 (Type N)

Insert one module into slot n in subunit A, the other into slot n + 1 in subunit B, where n is an even number between 0 and 30 (see section 6.2).
Set the selector switches of the modules in both subunits to P (P output).
Connect the 24 V DC supply voltage (insert jumper in subunit A only; connect unused terminals (3, 5, 7, 9) in subunit B to L +).
Wire the outputs in subunit B with the actuators (see Figure 5-17).
Initialize the outputs in DB1.

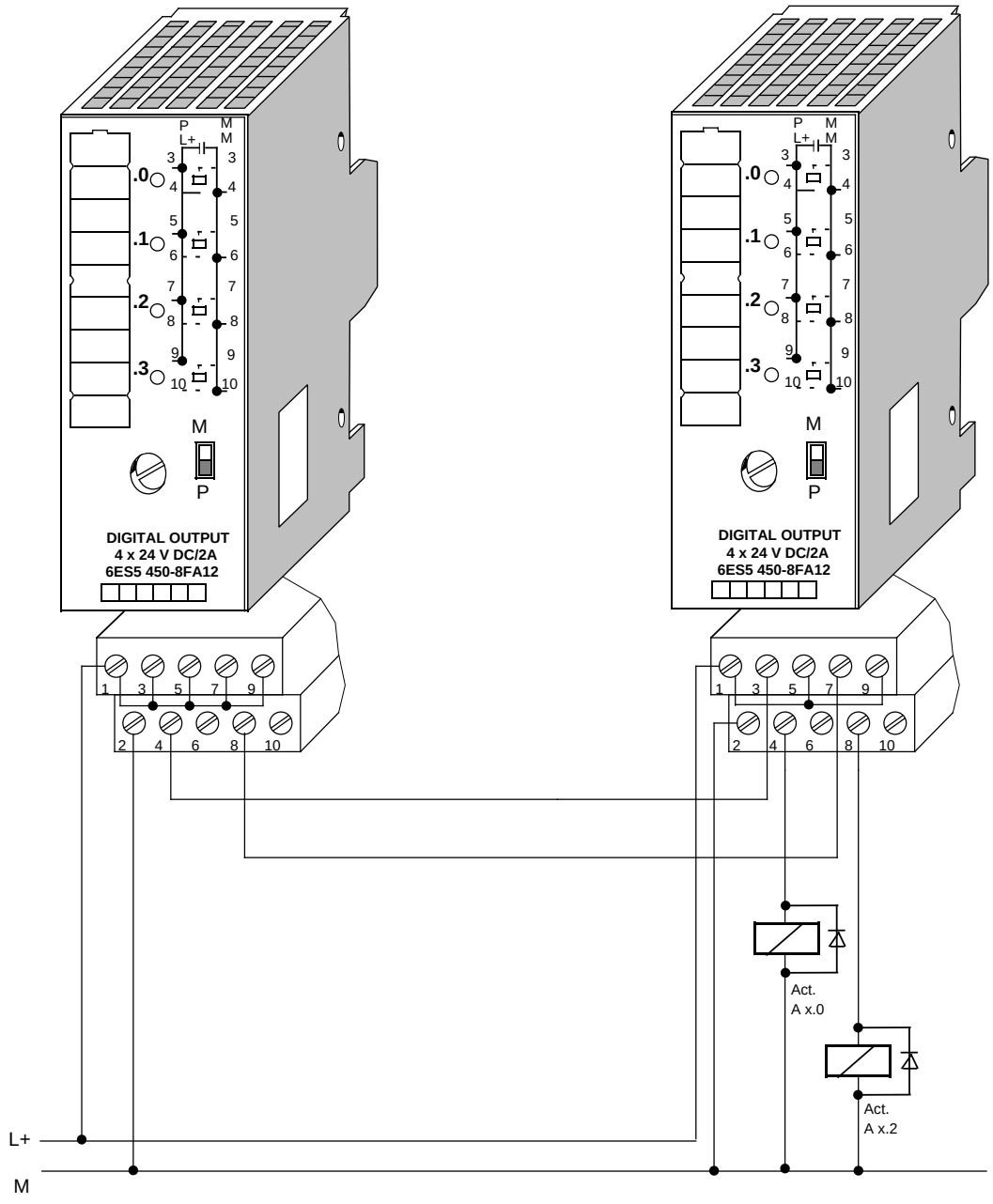


Figure 5-17. Example of a Connection for Single-Pole, Two-Channel Actuator Control

5.5.3 Expansion of Basic Unit with Failsafe Analog Value Processing

When the 464-8MG11 analog input module is used for safety-related tasks, it must always be redundant, that is, at least one module in subunit A and one in subunit B. Two or more 464-8MG11 analog input modules must be combined with one another, depending on the quality level and required availability.

Use of the AI 464-8MG11 in failsafe systems is permitted only in conjunction with integral function blocks FB 232 and FB 233 (see section 11.7).

Prerequisites for the Voltage Supply for Analog Sensors

The AI 464-8MG11 provides no voltage for analog sensors. You must therefore supply the voltage for the analog sensors from the 24 V DC source which also supplies the basic unit and the analog input module.

Prerequisites for the Process Signals from Analog Sensors

In order for the S5-95F to be able to detect a fault on the analog input module, the process signal must change frequently. In static mode, you must interrupt the sensor circuits at regular intervals at the user level. After a waiting time of 500 ms, the analog input signals must provide a value of "0", which does not occur under normal operating conditions.

Five hundred milliseconds after the interruption has been cleared, the user program can once again read/process the current analog value.

Example: Connecting Two Sensors for Quality Levels up to AK 4 to DIN V 19250

Insert one module into slot n in subunit A and a second module into slot n + 1 in subunit B, where n is an even number between 0 and 6 (see section 6.2).
Set the network frequency and the number of channels on the ”operating mode” switch (see section 11.3).
Wire the inputs with the sensors (see Figure 5-18).
Initialize function block FB 232 in the user program.

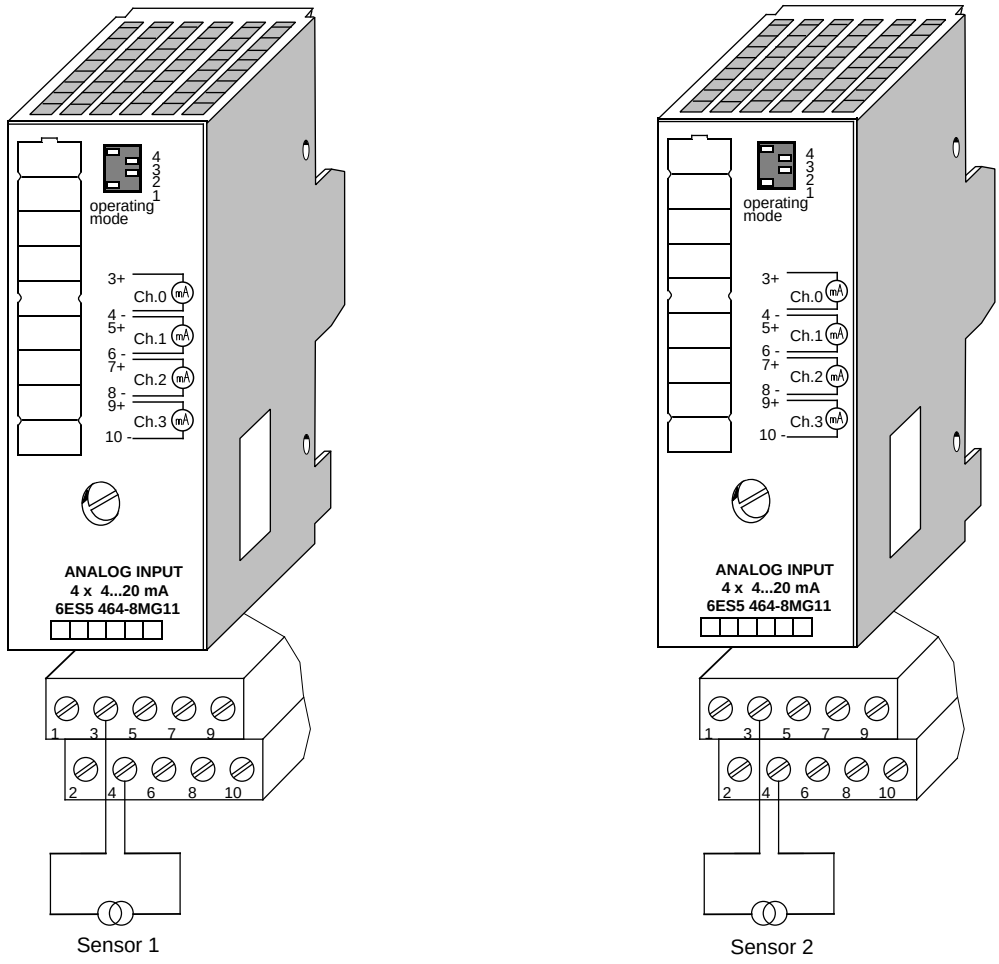


Figure 5-18. Example for Connection of Two Current Sensors

5.6 Connection of Non-Failsafe I/O Modules

When using standard S5-100U I/O modules, please note that these modules have a lower electromagnetic compatibility than failsafe modules; however, this does not have adverse effects on the safety functions of the overall system. An overview of modules which can be used in the S5-95F can be found in Appendix A.

Connecting Eight-Channel Digital Modules

These modules do not have a two-wire connection. You therefore need an external distribution block.

The eight channels of a module are numbered from .0 through .7. One terminal on the terminal block is assigned to each channel. The terminal assignment and the connection diagram are printed on the front plate of the module.

Connecting Eight-Channel Input Modules (Type A)

The sensors must be connected to terminal 1 via the L+ terminal block.

Example: Connecting a sensor to channel 4 (address I 3.4) on an input module in slot 3 (see Figure 5-19).

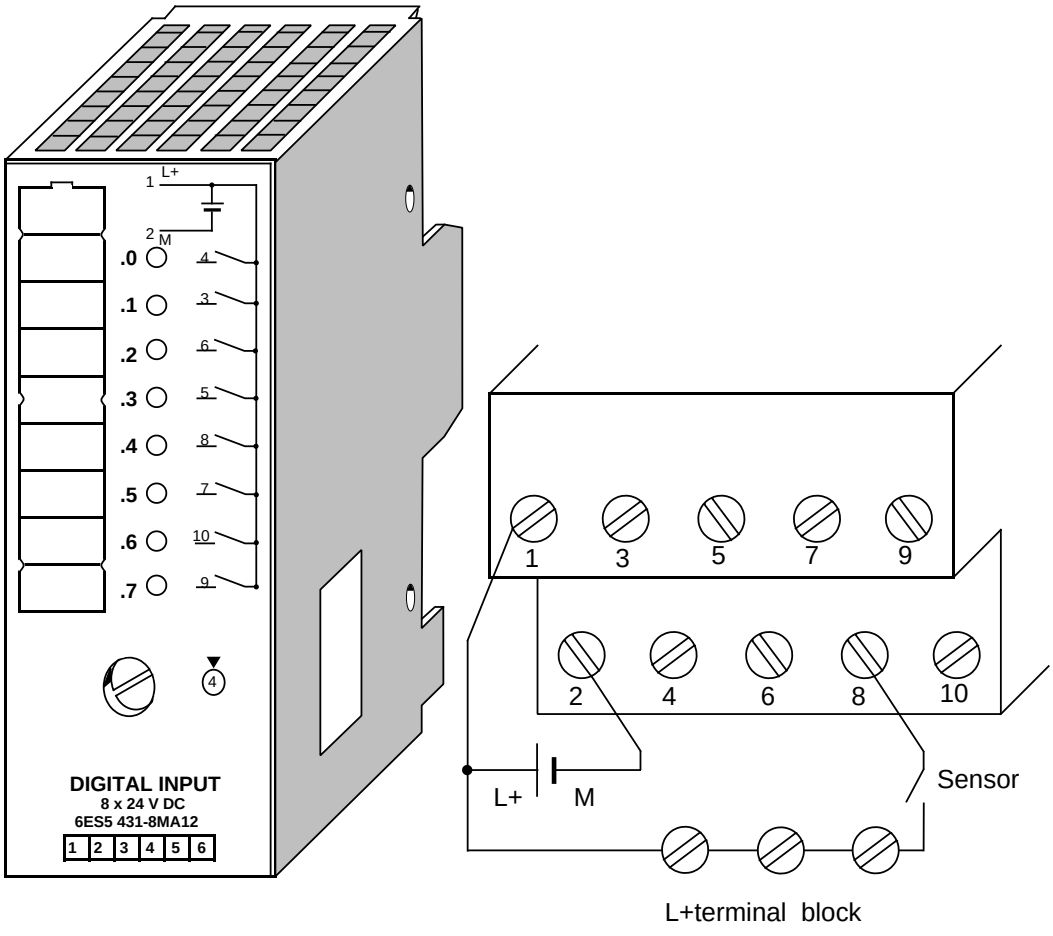


Figure 5-19. Connecting a Sensor to Channel 4

Connecting Eight-Channel Output Modules (Type J)

The actuators must be connected to terminal 2 via the M (negative) terminal block. This does not apply to the digital output module 8 × 5 to 24-V DC/0.1 A.

Example: Connecting a lamp to channel 6 (address Q 5.6) on an output module in slot 5 (see Figure 5-20).

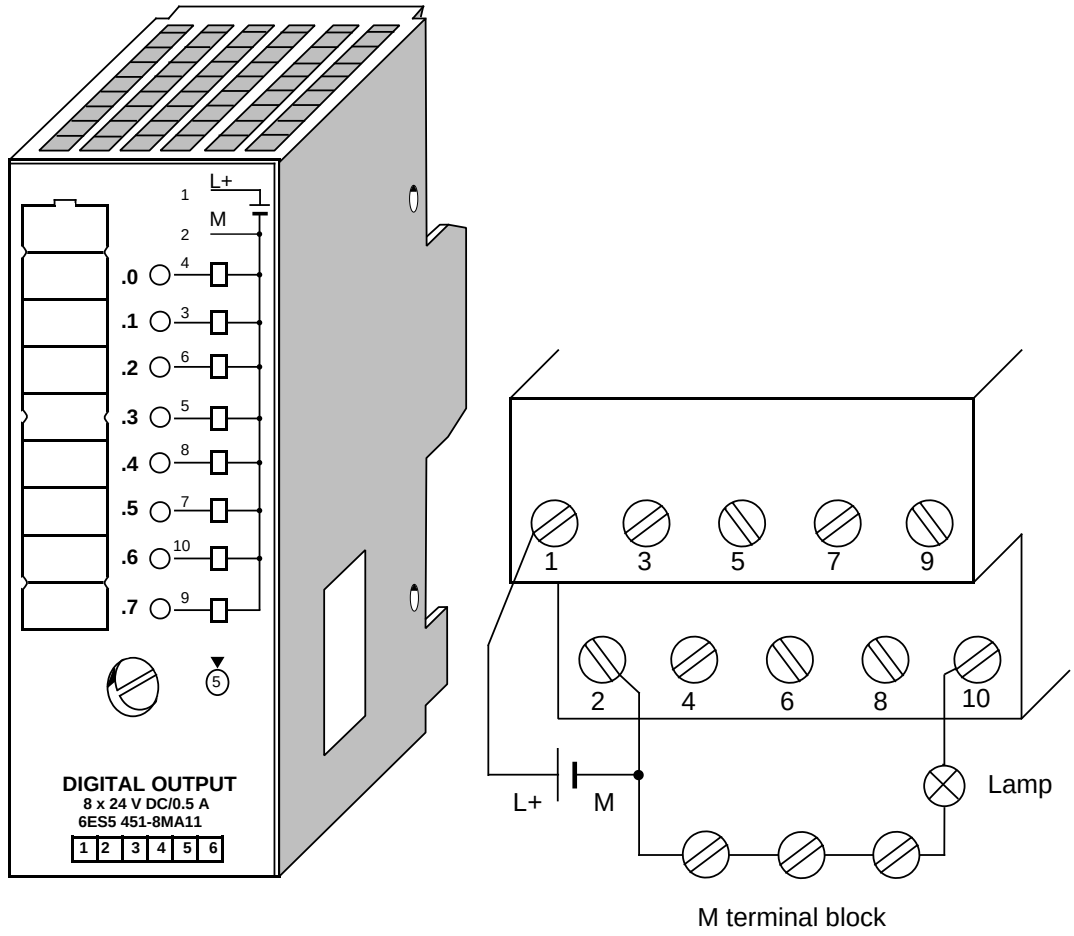


Figure 5-20. Connecting a Lamp to Channel 6

Connecting the Digital Input/Output Module DI/DQ 482 (Type A and J)

Use only slots 0 through 7 when you plug the module into the bus unit (4 slots in each subunit).

Use a 40-pin cable connector with a screw-type connection or crimp snap-in connection for wiring. The module does not have a two-wire connection. You must therefore use an external distribution block.

Every channel is assigned a terminal on the 40-pin connector. The channel numbers are printed on the front plate.

The 16 channels on the input side (IN) and on the output side (OUT) are numbered from n.0 through n.7 and from n+1.0 through n+1.7. "n" is the start address of the slot. Slot 0, for example, has the start address of n=64 (see Chapter 6).

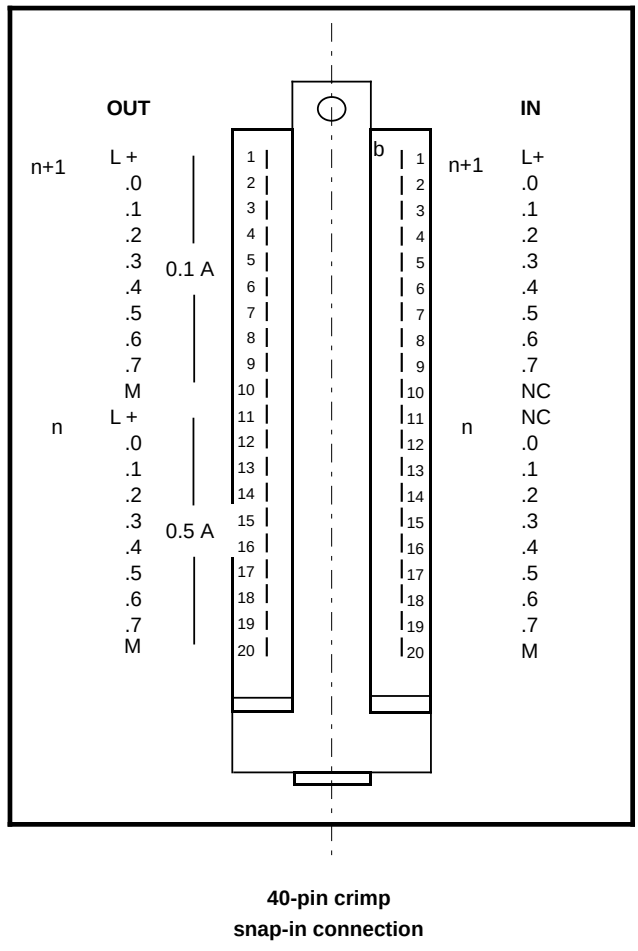


Figure 5-21. Front View of the Digital I/O Module with a Crimp Snap-In Connector (simplified view and not true to scale)

Example: The module is plugged into slot 0, its start address is 64.0 (see section 6.5). Inputs and outputs have the same address. A sensor is to be connected to input I 64.4 and a lamp to output Q65.3. Figure 5-22 illustrates the wiring on the front connector.

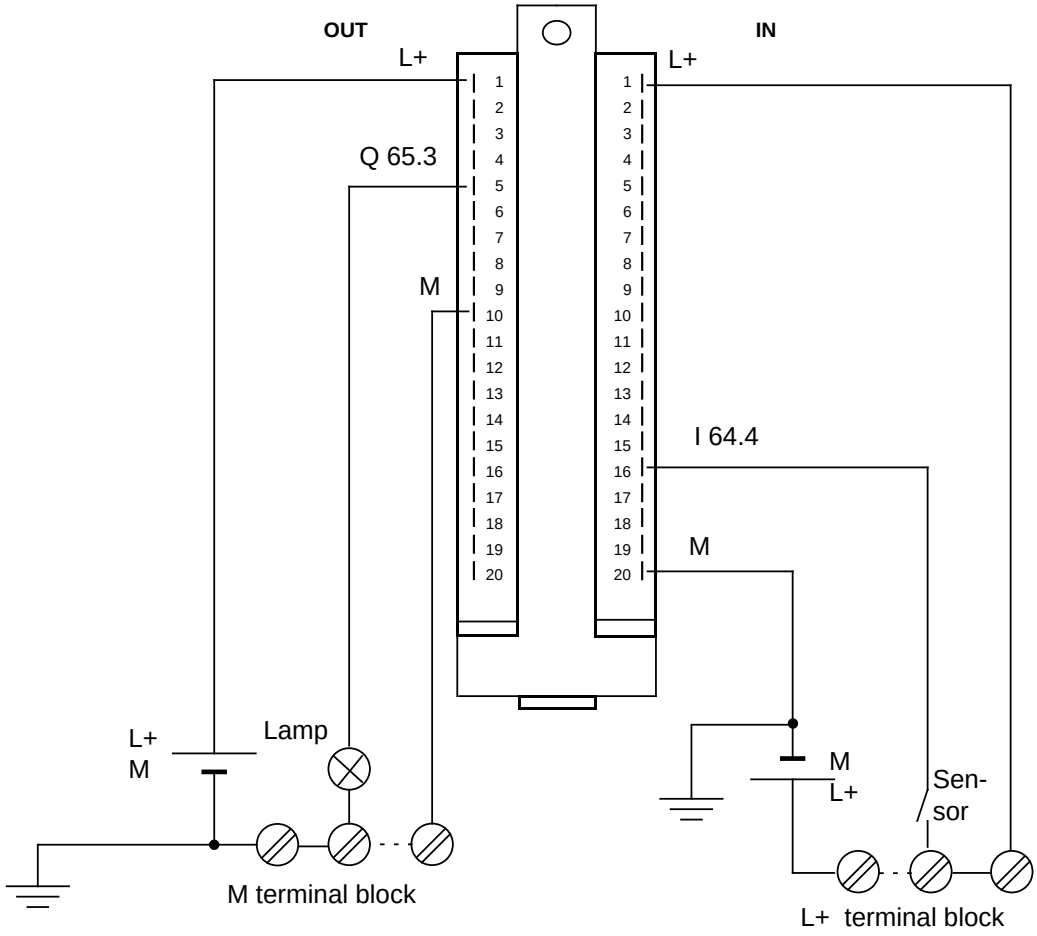


Figure 5-22. Connecting a Sensor and a Lamp to Digital Input/Output Module 482

5.7 Power Supplies for the S5-95F

The control for the S5-95F consists of several electrical circuits:

- Internal control circuits for the S5-95F (24 V DC)
- Control circuit for the onboard DI/DQ logic (24 V DC)
- Control circuit for the counter and interrupt logic (24 V DC)
- Sensor circuits (24 V DC for onboard I/Os) and
- Load circuits for the actuators (24 V DC for onboard I/Os)

Internal Control Circuits

The internal control circuits for the S5-95F have a common reference potential. They include: programmer interface, supply of the bus units and the control circuits for the external I/Os. The internal control circuits are supplied via the 24 V DC feeder on the basic unit.

Control Circuit for the Onboard DI/DQ Logic

The control circuit for the onboard DI/DQ logic has its own reference potential and is galvanically isolated from the other circuits. You must supply this circuit with 24 V DC.

Control Circuit for the Counter and Interrupt Logic

The control circuit for the counter and interrupt logic has its own reference potential and is galvanically isolated from the other circuits. You must supply this circuit with 24 V DC. A suppressor circuit is required for protection against surge pulses to IEC 801-5 (see section 3.4.5).

Sensor Circuits

The sensor circuits supply the sensors for the onboard and external digital input modules. Depending on the digital input module used, you can have 24 V DC, 120 V AC and 230 V AC sensor circuits (input circuits).

Load Circuits

The load circuits supply the actuators of the process I/Os. Depending on the digital output module used, you can have 24 V DC, 115 V AC and 230 V AC load circuits (output circuits).

Requirements for 24 V DC Supplies

You must supply all 24 V DC circuits from (electrically) isolated power sources to EN 60950 or from (electrically) isolated power sources to DIN VDE 0551 or EN 60742 and DIN VDE 0160 (see section 4.1.2).
Make sure that the voltage lies between 20 to 30 V (including ripple).
The AC ripple content must not exceed ±1.2 V.

> **Note**
>
> If you use a switched-mode power supply unit to supply floating analog modules and BEROs, then this supply must be filtered through a network.

5.8 Electrical Potentials for the Onboard and External I/Os

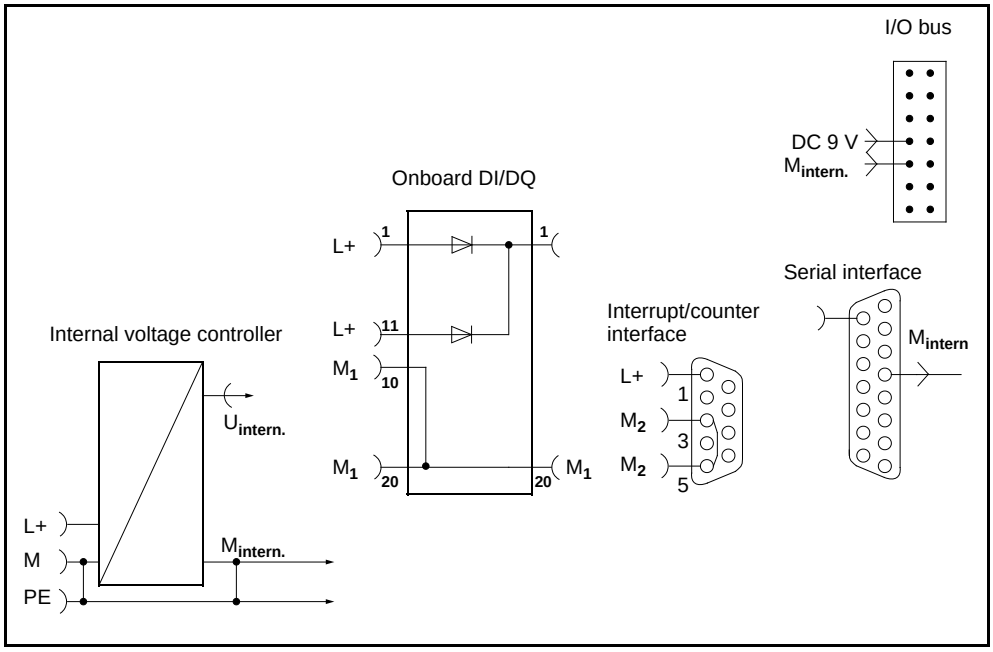
5.8.1 Onboard I/Os of the S5-95F Programmable Controller

The onboard I/Os of the S5-95F are galvanically isolated from the control circuit by optocouplers and allow a floating configuration.

The onboard I/Os are electrically subdivided into several groups:

- 1 group with 16 failsafe inputs
- 1 group with 8 failsafe outputs
- 1 group with 4 non-failsafe outputs
- 1 group with 4 failsafe interrupt inputs and 2 failsafe counters

Each group has its own 24 V DC connection.



$M_{intern.}$ = Reference potential of the internal logic

M_1 = Reference potential of the onboard DI/DQ

M_2 = Reference potential of the counter/interrupt inputs

Figure 5-23. Simplified Representation of the Electrical Potentials for the Basic Unit

5.8.2 Potential Bonding and Galvanic Isolation of External I/Os

The PLC is powered by its own control circuit. The I/Os are powered by the load circuit.

The circuits can be:

- Connected to the same grounding point (non-floating); or
- Galvanically isolated (floating)

Example of a Non-Floating Connection with Digital Modules

A 24-V DC load circuit has the same chassis grounding as the PLC's control circuit.

The common chassis grounding connection makes it possible for you to use reasonably priced non-floating I/Os. These modules function according to the following principles.

- Input modules: The M cable (control circuit chassis ground) is the reference potential. A voltage drop on cable affects the input signal level V_I.
- Output modules: Terminal 2 (M) of the terminal block is the reference potential. A voltage drop V_2 on cable raises the chassis ground of the output driver and thereby reduces the resulting control voltage V_{CV}.

Connection of Single-Channel Non-Floating External I/Os

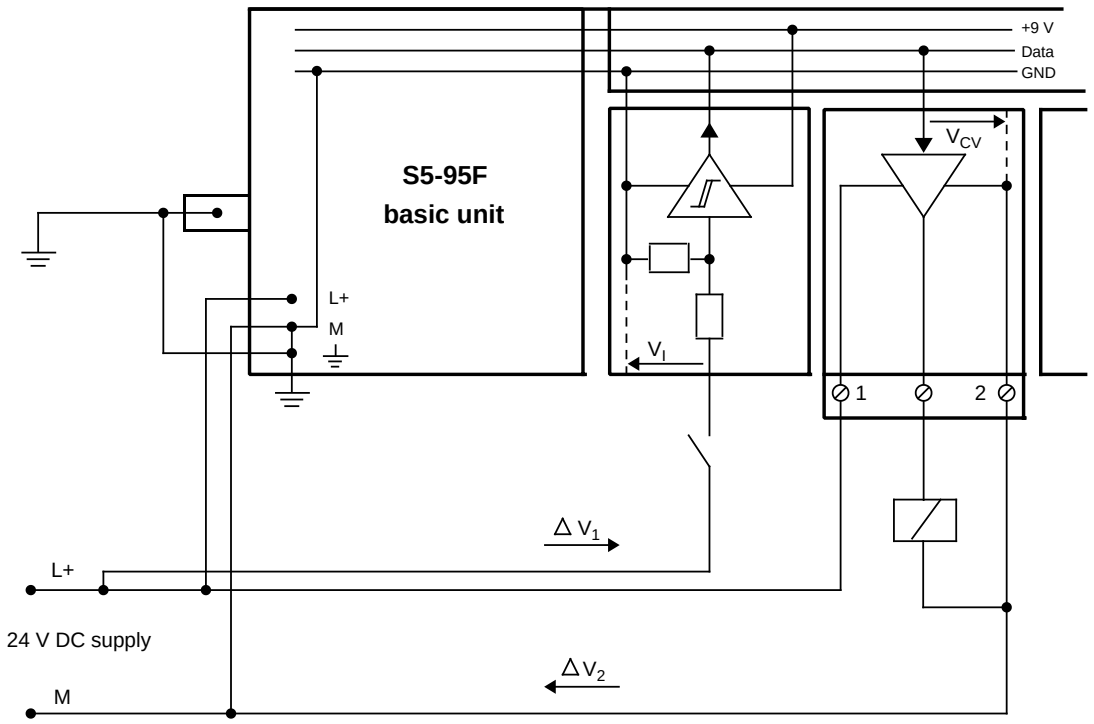


Figure 5-24. Simplified Representation of a Non-Floating, External I/O Connection

When you have a non-floating configuration, you must make certain that the voltage drop on cables and does not exceed 1 V. If 1 V is exceeded, the reference potentials could change and the modules could malfunction.

> **Warning**
>
> If you use non-floating I/O modules, you must provide an external connection between the chassis ground of the non-floating I/O module and the chassis ground of the CPU.

Example of a Floating Configuration with Digital Modules

Floating configuration is required in the following situations.

- When you need to increase interference immunity in the load circuits
- When load circuits cannot be interconnected
- When you have AC load circuits

If you have a floating configuration, the PLC's control circuit and the load circuit must be galvanically isolated.

Figure 5-25 shows a simplified connection of galvanically isolated I/Os.

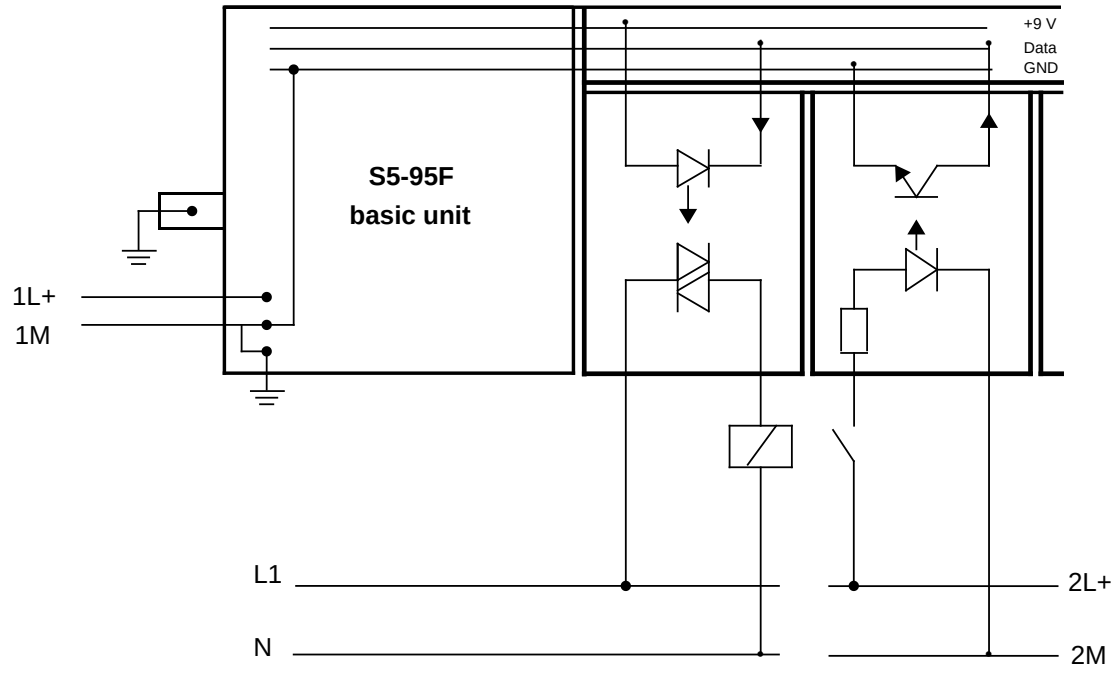


Figure 5-25. Simplified Representation of a Floating, External I/O Connection

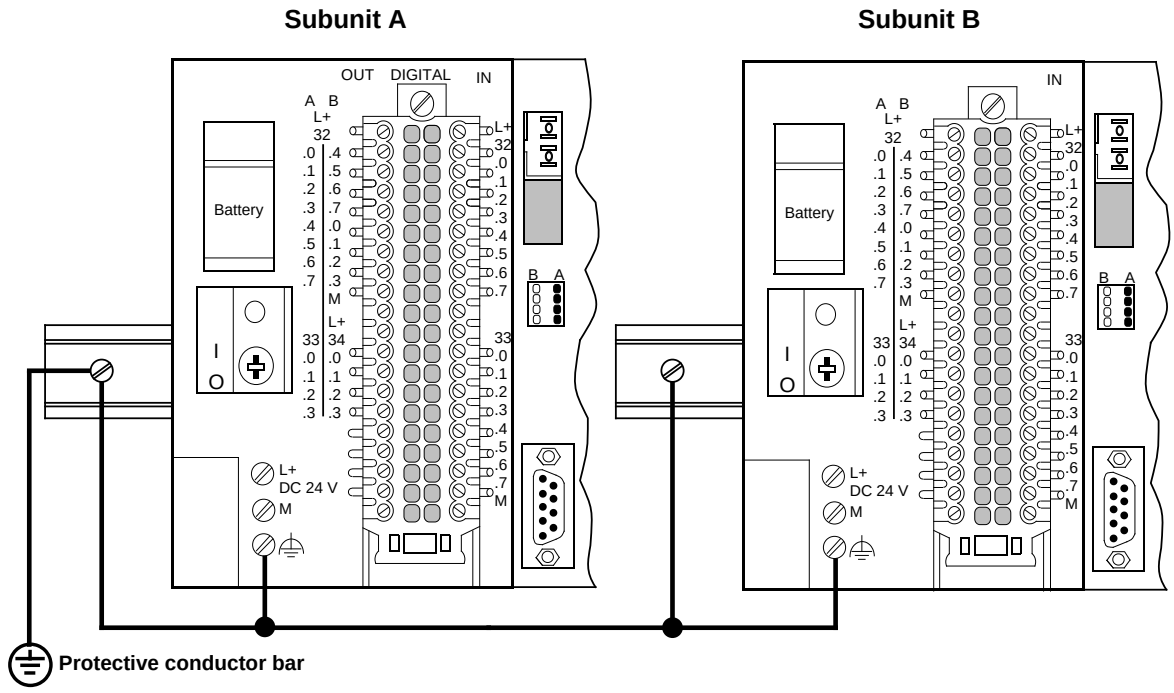
5.9 Grounded or Non-Grounded S5-95F Configuration

The S5-95F can be configured both with a grounded or with an isolated (non-grounded) reference potential.

Configuration with Grounded Reference Potential

Usually, you should configure the S5-95F with a grounded reference potential. The grounded configuration ensures a very high degree of interference immunity. Interference currents that occur are discharged by the standard mounting rail to the protective conductor.

Figure 5-26 shows the connections necessary for a grounded S5-95F configuration.



To configure the S5-95F with a grounded reference potential, you should proceed as follows:

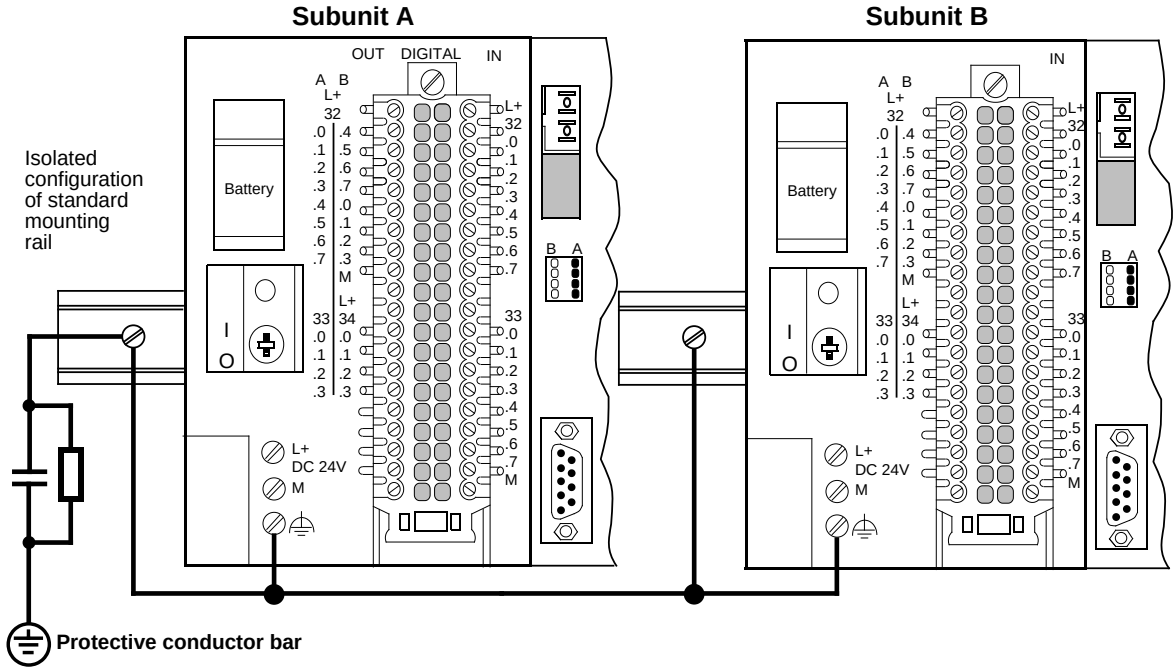
Connect the ⏚ terminal of subunit A and subunit B with the standard mounting rail.
Connect both subunits with each other.
Connect the standard mounting rail with the protective conductor.

Figure 5-26. Grounded S5-95F Configuration

Configuration with Non-Grounded Reference Potential

In large systems, the various process areas often have different earth potentials. If you supply all circuits with an electrically isolated functional extra-low voltage, you can operate an isolated S5-95F configuration - even with non-isolated input and output modules.
In the isolated configuration, interference currents that occur are discharged by the standard mounting rail via an RC network to the protective conductor. Grounding of the load voltages in an isolated configuration is also only permissible via an RC network.
For the isolated configuration, you require electrically isolated power supply units. The sensors and actuators must also fulfill the electrical isolation requirements.

Figure 5-27 shows the isolated configuration of an S5-95F programmable controller with capacitive connection to the protective conductor bar.



To configure the S5-95F with a non-grounded reference potential, you should proceed as follows:

Mount the standard mounting rails in an isolated configuration
Connect the ⏚ terminal of subunit A and subunit B with the standard mounting rail
Connect both subunits with each other
Connect the standard mounting rail with the protective conductor via an RC network (rating of the RC network: R = 100 k , C = 1 µF/500 V)

Figure 5-27. Non-Grounded S5-95F Configuration

> ⚠ **Attention**
>
> If you operate the S5-95F with a non-grounded reference potential, the configuration must be provided with an insulation monitor to detect double earth faults.

6 Addressing

6.1	Address Assignments for Onboard I/Os	6 - 2
6.2	Slot Numbering and Address Assignment for External I/Os	6 - 2
6.3	Digital Modules	6 - 4
6.4	Analog Modules	6 - 5
6.5	Combined Input and Output Modules	6 - 6
6.5.1	Digital Input/Output Module, 16 DI/16 DQ for 24 V DC	6 - 6
6.5.2	Function Modules	6 - 7
6.6	Structure of Process Input and Output Images	6 - 8
6.6.1	Accessing the Process Input Image	6 - 10
6.6.2	Accessing the Process Output Image	6 - 11
6.6.3	Direct Access to Onboard I/Os	6 - 12
6.6.4	Direct Access to External I/Os	6 - 13
6.7	Interrupt Process Image Tables and Time-Controlled Program Processing in OB13	6 - 13
6.7.1	Accessing the Interrupt PII	6 - 13
6.7.2	Accessing the Interrupt PIQ	6 - 14
6.8	Address Assignments in RAM	6 - 15

Figures

6-1 Address Assignment . . . . . 6 - 1
6-2 Consecutive Numbering of Slots in a Single-Tier Configuration . . . . . 6 - 2
6-3 Slot Numbering in a Multi-Tier Configuration . . . . . 6 - 3
6-4 Configuration of a Digital Address . . . . . 6 - 4
6-5 Address Assignment for Analog Modules . . . . . 6 - 5
6-6 Accesses to the PII . . . . . 6 - 10
6-7 Accesses to the PIQ . . . . . 6 - 11
6-8 Loading the Signal States of the Onboard Inputs and Outputs . . . . . 6 - 12
6-9 Accesses to the Interrupt PII . . . . . 6 - 14
6-10 Accesses to the Interrupt PIQ . . . . . 6 - 15

Tables

6-1 Address Assignment (Subunit A) . . . . . 6 - 6
6-2 Address Assignment (Subunit B) . . . . . 6 - 7
6-3 Structure of the Process Input Image (PII) and the Process Output Image (PIQ) . . . . . 6 - 8
6-4 S5-95F: Structure of the PII and PIQ . . . . . 6 - 9
6-5 RAM in the S5-95F . . . . . 6 - 16
6-6 Address Assignment in the System Data Area of the S5-95F . . . . . 6 - 18

6 Addressing

The two subunits of the S5-95F are provided with various types of inputs and outputs. The inputs and outputs located on the programmable controller itself are called onboard I/Os. The other I/Os are the failsafe, external I/Os specially developed for the S5-95F and the non-failsafe standard I/Os from the S5-100U programmable controller with which you can expand your S5-95F.

To access the inputs and outputs, you must assign them specific addresses.

The onboard I/O addresses cannot be changed. The external I/O addresses are slot-dependent.

Address Assignment for External I/O Modules

When you mount an external I/O module in a slot on a bus unit, the module is immediately assigned a slot number and consequently a fixed byte address in the process I/O images.

Connect the sensors and actuators either via a connector or to the terminal block of the bus units. The terminal selected determines the bit address (channel number) of the input or output.

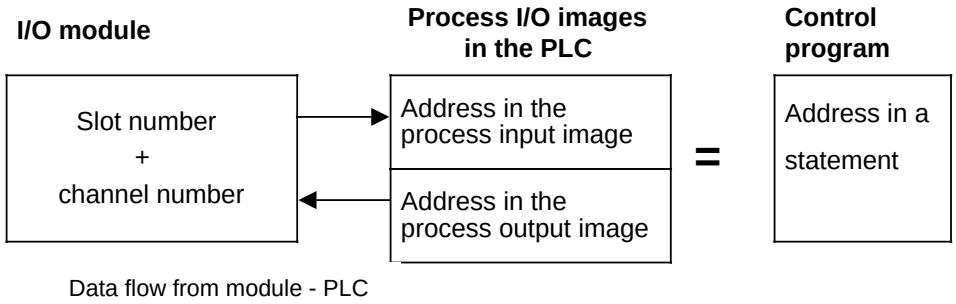


Figure 6-1. Address Assignment

Addressing of Single-Channel External I/Os

Single-channel I/Os are never failsafe. In the user program, these I/Os must always be referenced under the address under which they are used in the relevant subunit.

Addressing of Two-Channel, Failsafe External I/Os

Two-channel I/Os can be used only with the failsafe onboard I/Os or with redundant failsafe external I/O modules.

> **Note**
>
> You must always address failsafe external I/Os in the user program at the address where they are used in subunit A (failsafe external I/Os have always an even-numbered byte address)!

6.1 Address Assignments for Onboard I/Os

The addresses of the onboard I/Os are permanently assigned and cannot be changed by you. The assignment is shown in the following table:

Onboard I/Os	Byte/Word Address	Bit Address
Digital inputs, failsafe	IB 32 to 33	I 32.0 to I 33.7
Digital outputs, failsafe	QB 32	Q 32.0 to Q 32.7
Digital outputs, non-failsafe	QB 33 to 34	Q33.0 to Q 33.3 (subunit A) Q 34.0 to Q 34.3 (subunit B)
Counter A, failsafe	IW 36	-
Counter B, failsafe	IW 38	-
Interrupt inputs, failsafe	IB 59	I 59.0 to I 59.3

6.2 Slot Numbering and Address Assignment for External I/Os

With 8 bus units (16 slots), each subunit can have a maximum of four tiers.

The slots are assigned ascending numbers. Numbering begins with the slot next to the basic unit. Whether a module is plugged in or not has no effect on the numbering.

> **Note**
>
> The slots for subunit A are even-numbered and the slots for subunit B are odd-numbered.

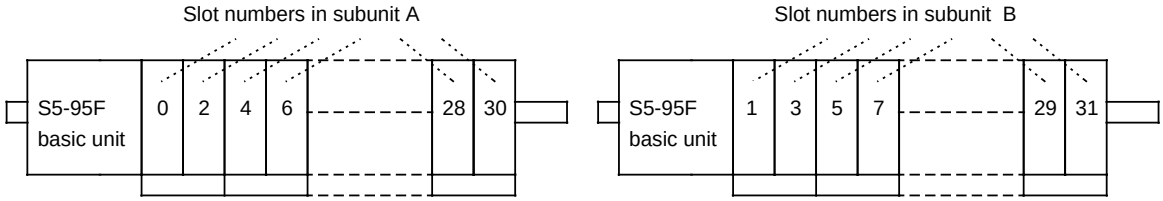


Figure 6-2. Consecutive Numbering of Slots in a Single-Tier Configuration

If the S5-95F consists of several tiers, numbering of the expansion tiers is continued with the slot on the extreme left.

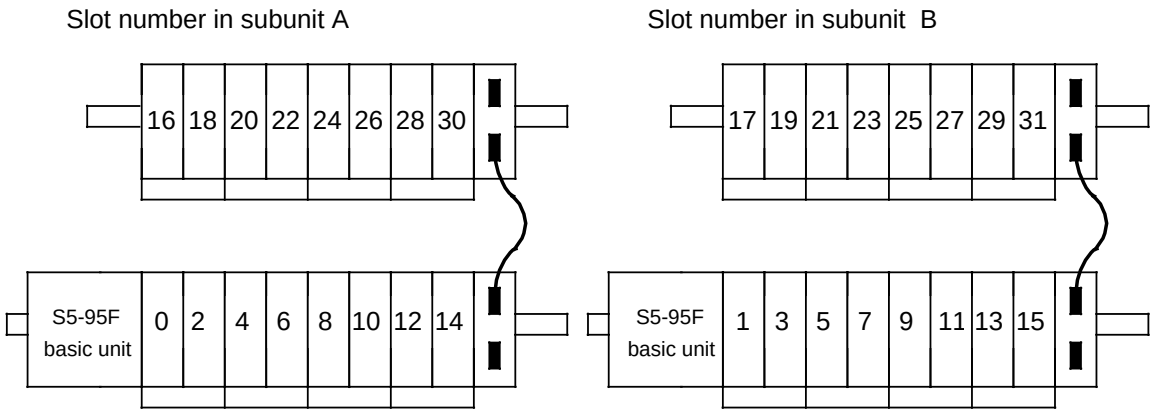


Figure 6-3. Slot Numbering in a Multi-Tier Configuration

When expanding your system, always add the new bus units to the topmost tier on the right. Otherwise, the slot numbers on the right of the new bus units will be changed, which requires address changes in your control program.

Note

After every expansion, check to make certain that the addressing used in the control program is the same as that in the actual configuration.

6.3 Digital Modules

Digital modules can be plugged into all slots.
Only two information states ("0" or "1", "OFF" or "ON") per channel can be transferred from or to a digital module.

Each channel of a digital module is displayed by a bit. This is the reason that every bit must be assigned its own number. Use the following form for a digital address:

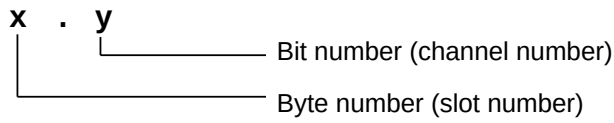


Figure 6-4. Configuration of a Digital Address

The address "**X.Y**" consists of two components:

Byte Address X (Slot Number X)

The byte address is the same as the number of the slot the module is plugged into.

Channel Number Y (Bit Address Y)

The channel number comes from the connection of the actuators or sensors to the terminals of the terminal block.
The assignment for the channel number and the terminal number is printed on the frontplate of the module.

Example: Address Assignment

You are connecting a 2-wire BERO proximity limit switch to an 8 x 24-V DC digital input module (6ES5 431-8MA12) at terminal 3. The other wire is routed to an L+ terminal block. The module is plugged into slot 3.

This defines the address used by the control program to evaluate the signal states of the BERO.

- The byte address is 3 since the module is plugged into slot 3.
- As shown on the frontplate, the terminals for channel number 1 are used.
- The complete address for the BERO switch is 3.1.

6.4 Analog Modules

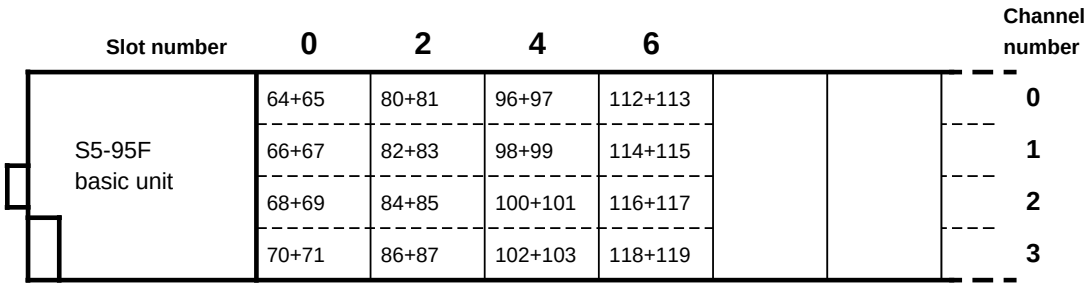
You can plug analog modules only into slots 0 through 7. Whereas only the information "0" or "1" ("ON" or "OFF") per channel can be transferred from or to a digital module (memory requirement 1 bit), it is possible to transfer 65,536 different items of information per channel from or to an analog module. The memory requirement is 16 bits=2 bytes=1 word.

The modules are addressed byte-by-byte or word-by-word with load or transfer operations.

The programmable controller takes this increased memory requirement into account when an analog module is plugged in.

- Eight bytes (=four words) are reserved per slot.
- The slot addressing area is changed.
- The permissible address space extends from byte 64 (slot 0, channel 0) to byte 127 (slot 7, channel 3).

Address assignments for analog modules in subunit A



Address assignments for analog modules in subunit B

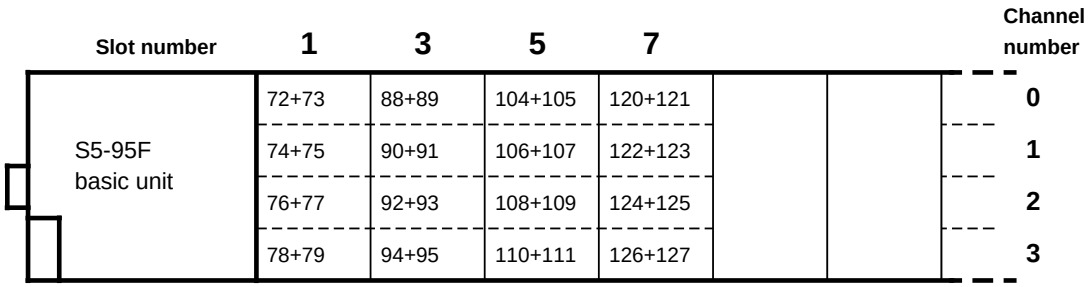


Figure 6-5. Address Assignment for Analog Modules

Examples:

1) Analog module in slot 2, channel number 0 bytes 80 and 81
2) Channel 1 address of an analog module in slot 5?
   Solution: bytes 106+107
   Bytes 92 and 93 Analog modules in slot 3, channel number 2

6.5 Combined Input Modules and Output Modules

With these modules it is possible to write data from the control program to the module and to read in data from the module to the control program.

The addresses in the process input image (PII) and process output image (PIQ) are identical.

6.5.1 Digital Input/Output Module, 16 DI/16 DQ for 24 V DC

The module can be used for non-safety-related tasks, and can be plugged into slots 0, 2, 4 and 6 of subunit A and slots 1, 3, 5 and 7 of subunit B.

The module occupies the same address space as an analog module. However, only the first two of the reserved bytes are used.

The address consists of byte address n or n + 1 and channel number Y. "n" is the start address of a slot, i.e. the first of the reserved bytes (e.g. byte 64 for slot 0). "n+1" is therefore the second of the reserved bytes. The designations "n" and "n + 1" are printed on the frontplate of the module.

The input and output data occupy the same addresses.

The channel number is defined by the connection of the actuators and sensors to the crimp connector.

The channel numbers are printed on the frontplate.

Table 6-1. Address Assignment (Subunit A)

Slot Number in Subunit A		0	2	4	6
Address PII (IN) and PIQ (OUT)	Channel n.0 to n.7	64.0 to 64.7	80.0 to 80.7	96.0 to 96.7	112.0 to 112.7
	Channel n+1.0 to n+1.7	65.0 to 65.7	81.0 to 81.7	97.0 to 97.7	113.0 to 113.7

Table 6-2. Address Assignment (Subunit B)

Slot Number in Subunit A		1	3	5	7
Address PII (IN) and PIQ (OUT)	Channel n.0 to n.7	72.0 to 72.7	88.0 to 88.7	104.0 to 104.7	120.0 to 120.7
	Channel n+1.0 to n+1.7	73.0 to 73.7	89.0 to 89.7	105.0 to 105.7	121.0 to 121.7

Examples: Determining the Address

1) You plugged the module into slot 4 and connected an actuator at byte n, channel 4. The address is 96.4.
2) Address 113.3 indicates a sensor or an actuator is connected at byte n + 1, channel 3. The module is plugged into slot 6.

6.5.2 Function Modules

Function modules have module-specific addressing.

Some function modules are addressed like digital modules, and other function modules are addressed like analog modules.

The addressing for each function module is explained in the corresponding manuals.

6.6 Structure of Process Input and Output Images

Data from inputs are stored in the process input image (PII). Data from outputs are stored in the process output image (PIQ).

The PII and the PIQ each have an area of 128 bytes in the RAM memory.

The PII and the PIQ have identical structures. They can be divided into four areas:

Table 6-3. Structure of the Process Input Image (PII) and the Process Output Image (PIQ)

Byte Address in the PII and PIQ	I/O Area	Slot Number
0 to 31	External I/O	0 to 31
32 to 39, 59	Onboard I/Os	
56 to 58 60 to 63	Diagnostic and communication bytes	
64 to 127	External I/O	0 to 7

- The address space for byte 0 through 31 is reserved for information from or to modules that are addressed like digital modules.

- The reserved address space for information from or to the onboard I/Os:
  - Byte 32 to 33 Failsafe digital inputs
  - Byte 32 Failsafe digital outputs
  - Byte 33 to 34 Non-failsafe digital outputs
  - Byte 35 Hardware diagnostic byte
  - Byte 36 to 39 Failsafe counters
  - Byte 56 Communication byte OB1 --> OB2
  - Byte 57 Indication of the occurred OB2 interrupts for evaluation in OB1
  - Byte 58 Indication of the interrupt bits reset in OB2 for evaluation in OB1
  - Byte 59 Failsafe OB2 interrupt inputs
  - Byte 60 Indication of the occurred OB2 interrupts for evaluation in OB2
  - Byte 61 to 63 Diagnostic byte for software interrupts

- The address space in bytes 64 to 127 is reserved for information from or to modules that are addressed like analog modules.
  The exact assignment of these four areas for the two programmable controllers is shown in the following tables.

Note

Reading from an empty slot always results in signal state "0".

Areas and their Addresses in the Process Image of the S5-95F

Table 6-4. S5-95F: Structure of the PII and PIQ

Relative Byte Addresses	I/O Area	PII	PIQ
0 31	Digital inputs and outputs of the external I/Os	6300_H . . $631F_H$	6380_H . . $639F_H$
32 34	Digital inputs and outputs of the onboard I/Os	6320_H 6321_H	$63A0_H$ . . $63A2_H$
35	Hardware diagnostic byte Configuring byte for short-circuit test and rotational speed monitoring	6323_H	$63A3_H$
36 37	Onboard counter A	6324_H . . 6325_H	. . .
38 39	Onboard counter B	6326_H . . 6327_H	. . .
40 55	Reserved for integral function blocks	6328_H . . 6337_H	$63A4_H$ . . $63BF_H$
56 58	OB2 communication bytes	6338_H . . $633A_H$	. . .
59	Onboard interrupt inputs (hardware interrupts)	$633B_H$	.
60 63	Diagnostic bytes for interrupts	$633C_H$ . . $633F_H$	. . .
64 127	Analog inputs and outputs of the external I/Os	6340_H . . $637F_H$	$63C0_H$ . . $63FF_H$

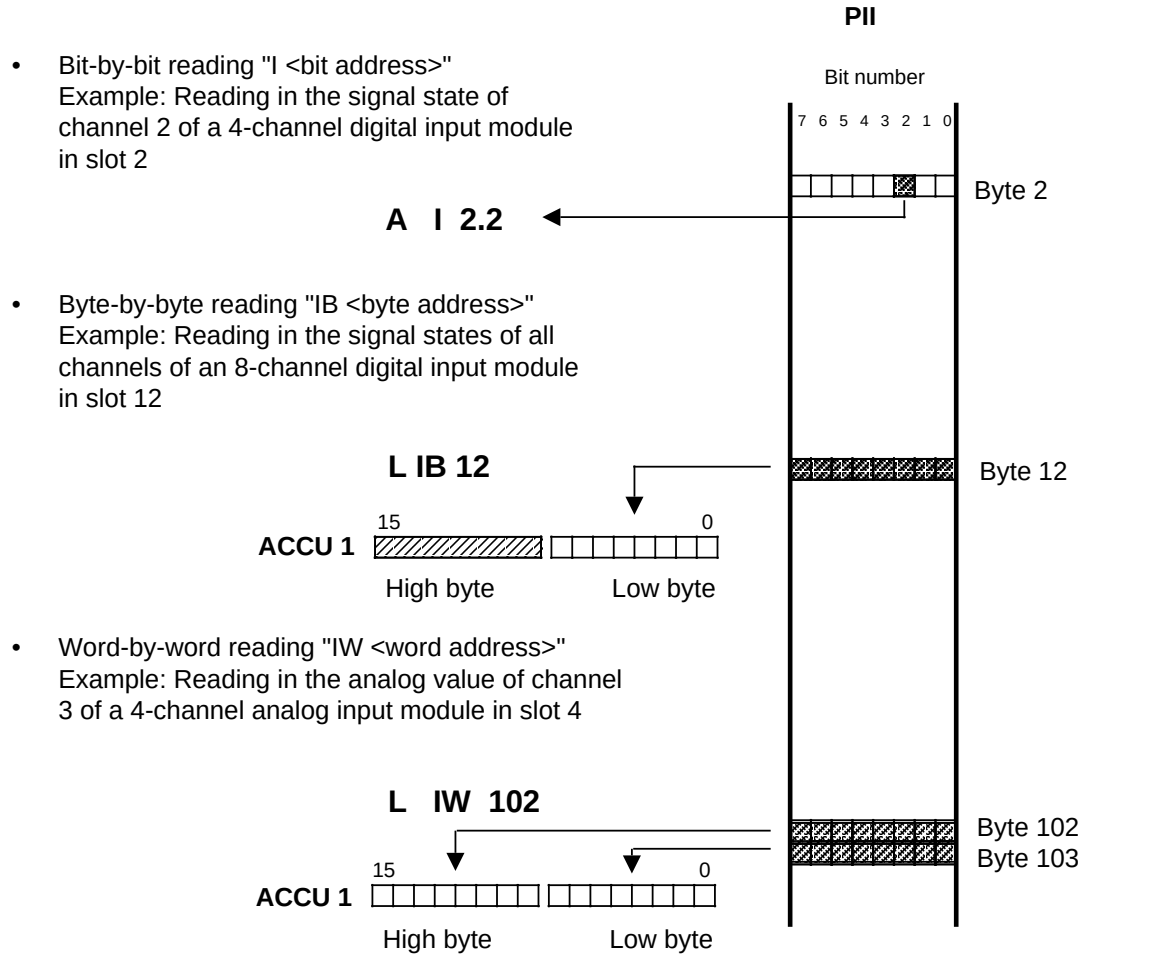
6.6.1 Accessing the Process Input Image (PII)

During a data cycle, data is read into the process input image (PII) from input modules.
This data is available to the control program for evaluation in the next program processing cycle.

Access to the PII is expressed by the operand identifiers "I", "IB", or "IW" in a statement in the control program.

The letter "L" identifies the "Load" operation (see Chapter 8). The letter "A" identifies the "AND logic" operation (see Chapter 8).

> **Note**
>
> Failsafe external I/Os may be accessed only via even-numbered byte addresses. Word-by-word accessing of failsafe external I/Os is rejected by the S5-95F with an error message.



- Bit-by-bit reading "I <bit address>"
  Example: Reading in the signal state of channel 2 of a 4-channel digital input module in slot 2
- Byte-by-byte reading "IB <byte address>"
  Example: Reading in the signal states of all channels of an 8-channel digital input module in slot 12
- Word-by-word reading "IW <word address>"
  Example: Reading in the analog value of channel 3 of a 4-channel analog input module in slot 4

Figure 6-6. Accesses to the PII

6.6.2 Accessing the Process Output Image (PIQ)

During a program cycle, data coming from the control program to the output modules is written into the process output image (PIQ). The data is transferred to the output modules in the following data cycle.

Access to the PIQ is expressed by the operand identifiers "Q", "QB", or "QW" in a statement in the control program.
The letter "T" identifies the "Transfer" operation (see Chapter 8). The "=" character assigns the result of a logic operation (RLO) to the operand that follows the character (see Chapter 8).

> **Note**
>
> Failsafe external I/Os may be accessed only via even-numbered byte addresses. Word-by-word accessing of failsafe external I/Os is rejected by the S5-95F with an error message.

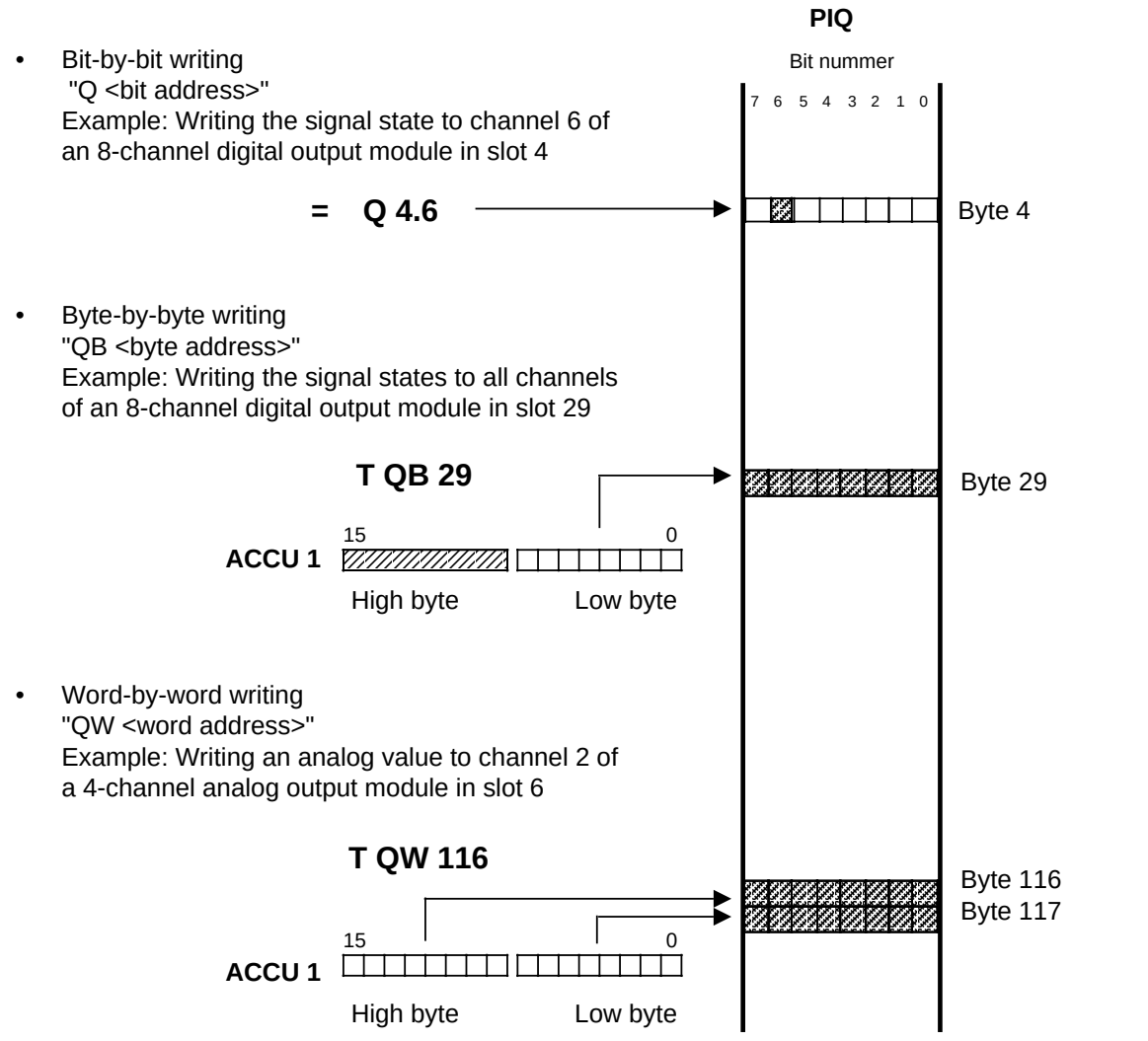


Figure 6-7. Accesses to the PIQ

6.6.3 Direct Access to Onboard I/Os

Direct access makes it possible for you to exchange information in the interrupt OBs with the I/Os without having to read/write the information from/to the PIQ or PII first. To a degree, you circumvent the PII or PIQ and directly exchange information with the inputs or outputs.

You have direct access to all of the onboard I/Os. Using the "L PB/PYxx" or "L PWxx" statements, you can read in (load) the signal state of an input byte or input word into the control program at any time. You can also use the "T PB/PYxx" or "T PWxx" statements to write (transfer) output information directly to an output byte or output word.

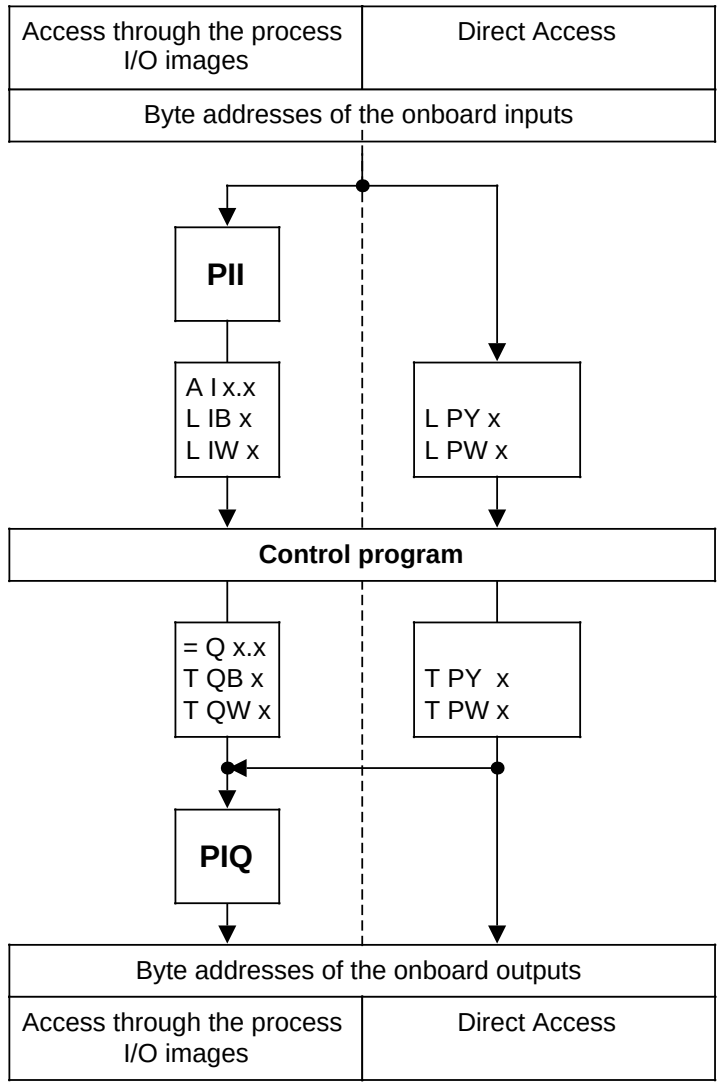


Figure 6-8. Loading the Signal States of the Onboard Inputs and Outputs

Note

Direct accessing of the external I/Os is permissible only in OB13 (see section 6.7). If you program direct accesses to external I/Os in OB1, OB2 or OB3, the S5-95F reacts with a corresponding error message and STOP.

6.6.4 Direct Access to External I/Os

External I/Os can be directly accessed only in OB 13.

Please note that the operation L PW 31 loads only the I/O byte 31 into the high byte of accu 1; the low byte is loaded with 0. The operation T PW 31 affects only byte 31.

6.7 Interrupt Process Images and Time-Controlled Program Processing in OB13

The S5-95F makes time-controlled program processing possible. OB13 must be programmed for this purpose. OB13 is called up periodically. You can determine the call up interval.

Within time-controlled program processing, you can use the operand identifiers, "PB", "PY", and "PW", not only with the onboard I/Os, but also with the external I/Os.

Load and transfer operations act as direct access (see section 6.6.3) to the onboard I/Os if these operand identifiers are used.

However, if the statement is in reference to the external I/Os, the programmable controller does not directly access the external I/O modules. It accesses the interrupt process images.

- The interrupt process images are used only in time-controlled program processing.
- The interrupt process images and the "normal" process images have an identical structure.
- The interrupt process input image (interrupt PII) and the interrupt process output image (interrupt PIQ) each have an area of 128 bytes in RAM.

Note

Access to the interrupt process images (I/O) can only be byte-by-byte or word-by-word. Failsafe external I/Os may be accessed only via even-numbered byte addresses. Word-by-word accessing of failsafe external I/Os is rejected by the S5-95F with an error message.

6.7.1 Accessing the Interrupt PII

- The interrupt PII can only be accessed within time-controlled program processing.
- The data from inputs is read into the interrupt PII only at the beginning of time-controlled program processing.
  The data is available exclusively to the time-controlled program for evaluation.

Time-Controlled Program Processing

Access to the interrupt PII is expressed by the "PB" or "PW" operand identifiers in a statement in a time-controlled program.
The letter "L" represents the "Load" operation (see Chapter 8).

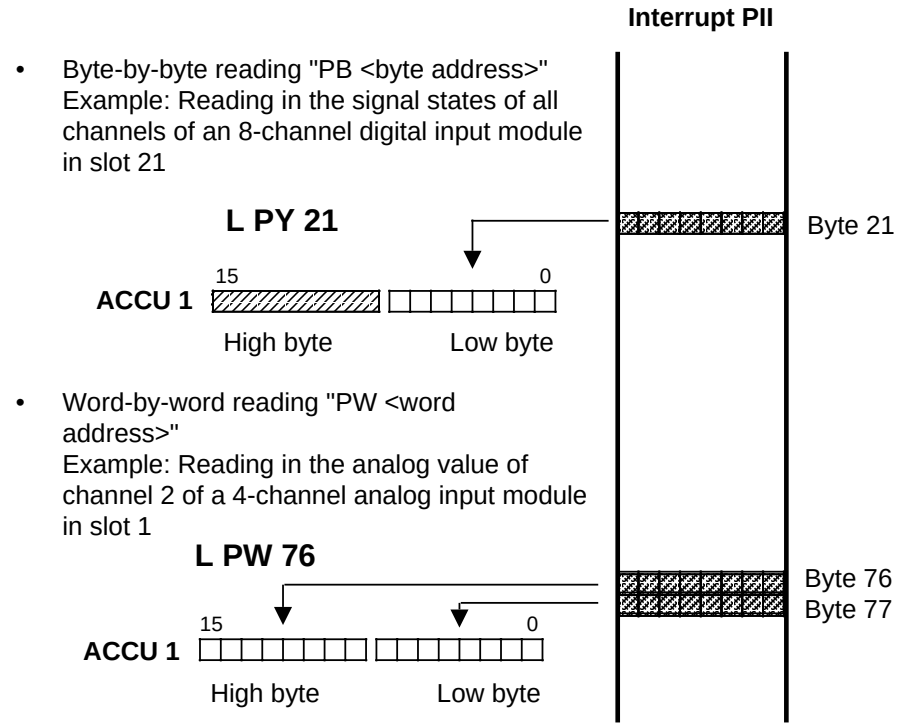


Figure 6-9. Accesses to the Interrupt PII

6.7.2 Accessing the Interrupt PIQ

- Data can be written to the interrupt PIQ only within time-controlled program processing. Use the transfer operations T PY/T PW.
- Data from the time-controlled program to outputs (via T PY/T PW) are written during time-controlled program scanning to both the ”normal” PIQ and to the interrupt PIQ as long as the outputs in question are those of external I/Os.
- Data from the interrupt PIQ are read out to the outputs in the next interrupt output data cycle.
- When OB1 has been processed, the operating system updates the interrupt PIQ by copying the data in the PIQ to the interrupt PIQ (interrupt PIQ is overwritten).

Note

The interrupt output data cycle is executed only after the interrupt PIQ has been written to.

Access to the interrupt PIQ is expressed by the "PB" or "PW" operand identifiers in a statement in the time-controlled program.
The letter "T" identifies the "Transfer" operation (see Chapter 8).

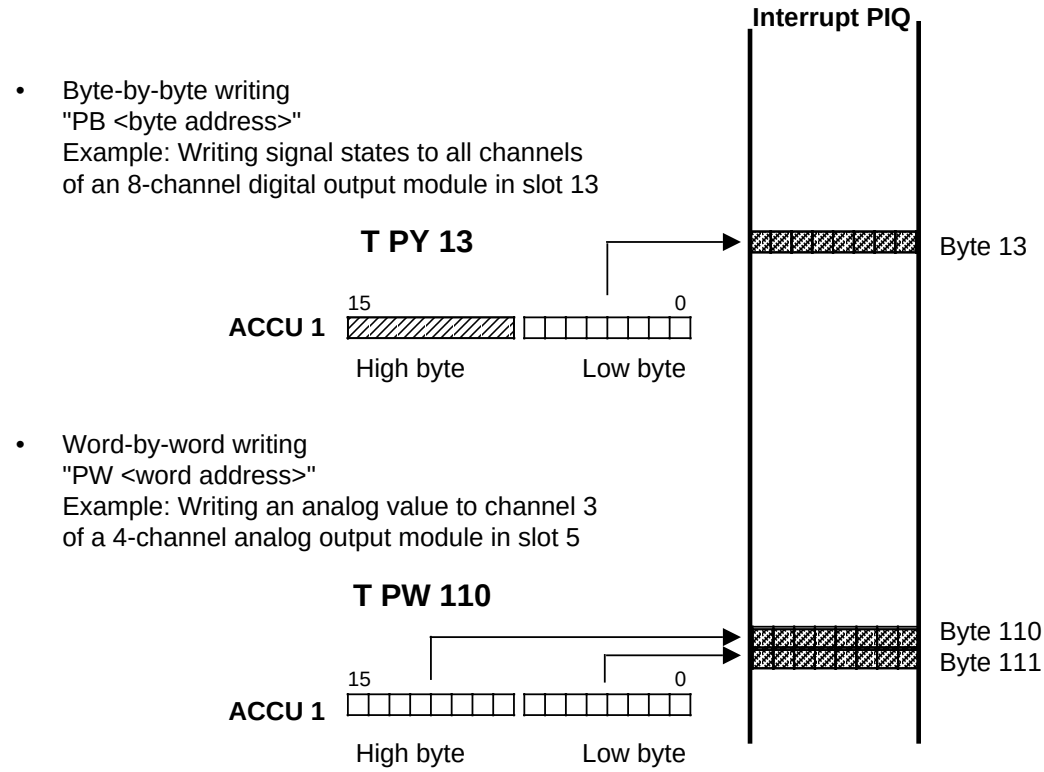


Figure 6-10. Accesses to the Interrupt PIQ

6.8 Address Assignments in RAM

The following tables contain information about the RAM of the two programmable controllers.

RAM in the S5-95F (see Table 6-5)
System data area of the S5-95F (see Table 6-6)

Table 6-5. RAM in the S5-95F

Address	RAM in the S5-95F
0000_H : $00FF_H$	Internal data
0100_H : $40FF_H$	Program memory
4100_H : $5CFF_H$	Internal data
$5D00_H$: $5EFF_H$	System data
$5F00_H$: $5FFF_H$	Timers T
6000_H : $600F_H$	Counters C (retentive)
6010_H : $60FF_H$	Counters (non-retentive)
6100_H : $613F_H$	Flags F (retentive)
6140_H : $61FF_H$	Flags F (non-retentive)
Address	**Interrupt Process I/O Images**
6200_H : $621F_H$	Digital interrupt PII (external I/Os)
6220_H : $623F_H$	Internal data
6240_H : $627F_H$	Analog interrupt PII (external I/Os)
6280_H : $629F_H$	Digital interrupt PIQ (external I/Os)
$62A0_H$: $62BF_H$	Internal data
$62C0_H$: $62FF_H$	Analog interrupt PIQ (external I/Os)
Address	**Process I/O Images**
6300_H : $631F_H$	Digital PII (external I/Os)
6320_H : 6321_H	Digital PII (onboard I/Os)
6322_H	Unassigned
6323_H	Hardware diagnostic byte
6324_H : 6327_H	PII counters A and B (onboard)
6328_H : 6337_H	Reserved for integral function blocks
6338_H $633A_H$	OB2 communication bytes
$633B_H$	PII interrupt inputs (onboard)
$633C_H$ $633F_H$	Diagnostic bytes for interrupts
6340_H : $637F_H$	Analog PII (external I/Os)
6380_H : $639F_H$	Digital PIQ (external I/Os)
$63A0_H$: $63A2_H$	Digital PIQ (onboard I/Os)
$63A3_H$	Configuring byte for short-circuit test and rotational speed monitoring
$63A4_H$ $63BF_H$	Reserved for integral FBs

Table 6-5. RAM in the S5-95F (continued)

Address	Process I/O Images
$63C0_H$: $63FF_H$	Analog PIQ (external I/Os)
6400_H : $75FF_H$	Internal data
Address	**Block Address List**
7600_H : $77FF_H$	OB
7800_H : $79FF_H$	FB
$7A00_H$: $7BFF_H$	PB
$7C00_H$: $7DFF_H$	SB
$7E00_H$: $7FFF_H$	DB
8000_H : $FFFF_H$	Internal data

The following table is a sequential listing of important system data.

Table 6-6. Address Assignment in the System Data Area of the S5-95F

System data word	Address (hex.)	Meaning
8 - 12	5D10 : 5D19	Integral real time clock clock data area, status word, error messages, compensation value
33	5D42 5D43	Level indicator for STEP 5 blocks (First free memory location in the internal RAM)
34	5D44 5D45	Level indicator for assembler blocks (First free memory location in the internal RAM)
35	5D46 5D47	Start address for STEP 5 blocks in the internal RAM
37	5D4A 5D4B	End address for STEP 5 blocks in the internal RAM
40 - 47	5D50 5D5F	CPU version/software version
128 - 159	5E00 5E3F	Block stack
203 - 214	5E96 5EAD	Interrupt stack



Warning

The above system data should only be used by persons familiar with the system. All non-specified system data are used by the operating system and should in no case be used by you. Improper use of the system data may result in malfunctioning of the S5-95F and can thus impair the system's failsafety.

7 Introduction to STEP 5

7.1	Writing a Program	7 - 1
7.1.1	Methods of Representation	7 - 1
7.1.2	Operand Areas	7 - 3
7.1.3	Circuit Diagram Conversion	7 - 3
7.2	Program Structure	7 - 4
7.2.1	Linear Programming	7 - 4
7.2.2	Structured Programming	7 - 5
7.3	Block Types	7 - 7
7.3.1	Organization Blocks (OBs)	7 - 9
7.3.2	Program Blocks (PBs)	7 - 11
7.3.3	Sequence Blocks (SBs)	7 - 11
7.3.4	Function Blocks (FBs)	7 - 11
7.3.5	Data Blocks (DBs)	7 - 16
7.4	Program Processing	7 - 18
7.4.1	START-UP Program Processing	7 - 19
7.4.2	Cyclical Program Processing	7 - 21
7.4.3	Maximum Response Time with Cyclical Program Processing	7 - 23
7.4.4	Time-Controlled Program Processing	7 - 25
7.4.5	Maximum Response Time with Time-Controlled Program Processing	7 - 28
7.4.6	Interrupt-Driven Program Processing	7 - 29
7.5	Processing Blocks	7 - 30
7.5.1	Changing Programs	7 - 30
7.5.2	Changing Blocks	7 - 30
7.5.3	Compressing the Program Memory	7 - 30
7.6	Number Representation	7 - 31

Figures

7-1	Compatibility of STEP 5 Methods of Representation	7 - 2
7-2	Nesting	7 - 6
7-3	Structure of a Block Header	7 - 8
7-4	Example of Organization Block Use	7 - 10
7-5	Programming a Function Block Parameter	7 - 13
7-6	Programming a Function Block	7 - 16
7-7	Data Block Contents	7 - 17
7-8	Validity Areas of Data Blocks	7 - 17
7-9	Setting the Start-Up Procedure	7 - 20
7-10	Cyclical Program Processing	7 - 21
7-11	Response Time of the Onboard I/Os (Worst Case)	7 - 23
7-12	Response Time of the External I/Os (Worst Case)	7 - 24
7-13	Call-Up Interval and Delay of OB13 Processing	7 - 25
7-14	Response Time with Time-Controlled Program Processing	7 - 29
7-15	Compressing the Program Memory	7 - 30
7-16	Bit Assignment of a 16-Bit Fixed-Point Binary Number	7 - 31
7-17	BCD and Decimal Formats	7 - 32

Tables

7-1	Comparison of Operation Types	7 - 2
7-2	Comparison of Block Types	7 - 7
7-3	Overview of Organization Blocks	7 - 9
7-4	Block Parameter Types and Data Types with Permissible Actual Operands	7 - 14
7-5	Reference Values for the OB13 Call-Up Delay (No Use of SINEC L1)	7 - 26
7-6	Comparison of Number Formats	7 - 32

7 Introduction to STEP 5

This chapter explains how to program the S5-95F. It describes how to write a program, how the program is structured, the types of blocks the program uses, and the number representation of the STEP 5 programming language.

7.1 Writing a Program

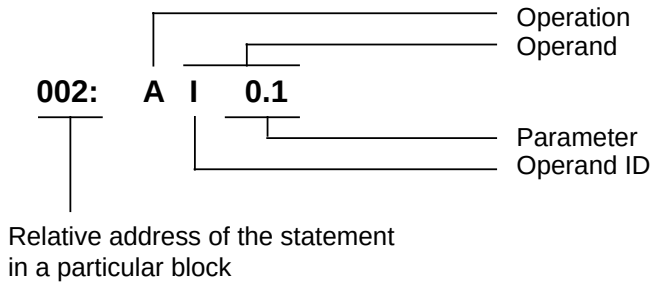
A control program specifies a series of operations that tell the programmable controller how it has to control a system. You must write the program in a very special language and according to specific rules so that the programmable controller can understand it. The standard programming language that has been developed for the SIMATIC S5 family is called STEP 5.

The S5-95F is designed for failsafe operation. That is why you can change some parameters in safety operation only via a special data block (see section 7.3.5). In the test mode, you must not observe this restriction.

7.1.1 Methods of Representation

The following methods of representation are possible with the STEP 5 programming language.

- **Statement List (STL)**
  STL represents the program as a sequence of operation mnemonics. A statement has the following format.



  The operation tells the programmable controller what to do with the operand. The parameter indicates the operand address.

- **Control System Flowchart (CSF)**
  CSF represents logic operations with graphics symbols.

- **Ladder Diagram (LAD)**
  LAD graphically represents control functions with circuit diagram symbols.

You cannot use CSF and LAD with the PG 605 and PG 615 programmers.

Each method of representation has its own special characteristics. A program block that has been programmed in STL cannot necessarily be output in CSF or LAD. The three methods of graphic representation are not compatible. However, programs in CSF or LAD can always be converted to STL. Figure 7-1 illustrates these points in a diagram.

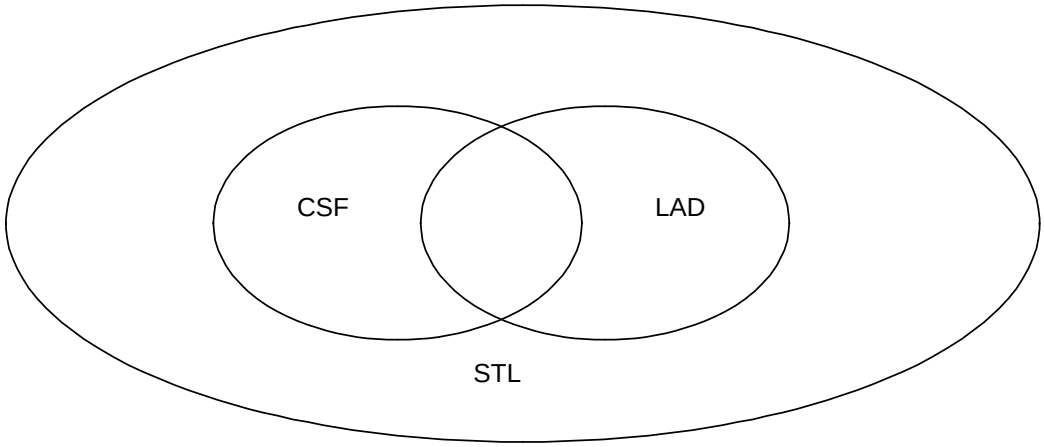


Figure 7-1. Compatibility of STEP 5 Methods of Representation

The STEP 5 programming language has the following three operation types.

- Basic
- Supplementary
- System

Table 7-1 provides further information on these operations.

Table 7-1. Comparison of Operation Types

STEP 5 PROGRAMMING LANGUAGE			
	Basic Operations	**Supplementary Operations**	**System Operations**
Application	In all blocks	Only in function blocks	Only in function blocks
Methods of representation	STL, CSF, LAD	STL	STL
Special features			For users with good system knowledge

Refer to Chapter 8 for a description of all operations and programming examples.

7.1.2 Operand Areas

The STEP 5 programming language has the following operand areas.

I	(inputs)	Interfaces from the process to the programmable controller
Q	(outputs)	Interfaces from the programmable controller to the process
F	(flags)	Memory for intermediate results of binary operations
D	(data)	Memory for intermediate results of digital operations
T	(timers)	Memory for implementing timers
C	(counters)	Memory for implementing counters
P	(peripherals)	Interfaces from the process to the programmable controller
K	(constants)	Defined numeric values
OB, PB, SB FB, DB	(blocks)	Program structuring aids

Refer to Appendix A for a listing of all operations and operands.

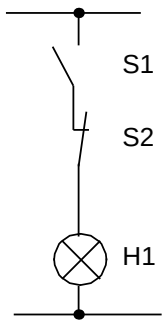
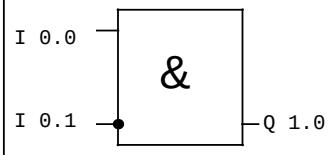
7.1.3 Circuit Diagram Conversion

If your automation task is in the form of a circuit diagram, you must convert it to STL, CSF, or LAD.

Example: **Hard-Wired Control**
A signal lamp (H1) is supposed to light up when a normally open contact (S1) is activated and a normally closed contact (S2) is not activated.

Programmable Control
The signal lamp is connected to an output (e.g. Q 1.0). The signal voltages of the two contacts are connected to two PLC inputs (e.g. I 0.0 and I 0.1).

The PLC scans to see if the signal voltages are present (signal state ”1” at the activated normally open contact or non activated normally closed contact). Both signal states are combined through logic AND. The result of logic operation (RLO) is assigned to output 1.0 (the lamp lights up).

Circuit Diagram	STL	CSF	LAD
	A I 0.0 AN I 0.1 = Q 1.0		

7.2 Program Structure

A program can be of linear or structured type.
Sections 7.2.1 and 7.2.2 describe these program types.

7.2.1 Linear Programming

Programming individual operations in one section (block) is sufficient for handling simple automation jobs.
For the programmable controllers, this is organization block 1 (OB1) (see section 7.3.1). The PLC scans this block cyclically (i.e., after it scans the last statement, it goes back to the first statement and begins scanning again).

Please note the following:

- When OB1 is called, five words are assigned to the block header in the program memory (see section 7.3).
- Normally, a statement takes up one word in the program memory.
  Two-word statements also exist (e.g., with the operation ”Load a constant”). Count these statements twice when calculating the program length.
- Like all blocks, OB1 must be terminated by a Block End statement (BE).

7.2.2 Structured Programming

To solve complex tasks, it is advisable to divide a program into individual, self-contained program parts (blocks).

This procedure has the following advantages.

- Simple and clear programming, even for large programs
- Capability to standardize program parts
- Easy alteration
- Simple program test
- Simple start-up
- Subroutine techniques (block call from different locations)

The STEP 5 programming language has the following five block types.

- **Organization Block (OB)**
  Organization blocks manage the control program.

- **Program Block (PB)**
  Program blocks arrange the control program according to functional or technical aspects.

- **Sequence Block (SB)**
  Sequence blocks are special blocks that program sequence controls. They are handled like program blocks.

- **Function Block (FB)**
  Function blocks are special blocks for programming frequently recurring or especially complex program parts (e.g., reporting and arithmetic functions). You can assign parameters to them and they have an extended set of operations (e.g., jump operations within a block).

- **Data Block (DB)**
  Data blocks store data needed to process a control program. Actual values, limiting values, and texts are examples of data.

The program uses block calls to exit one block and jump to another. You can therefore nest program, function, and sequence blocks (see section 7.3) randomly in up to 16 levels.

> **Note**
>
> When calculating the nesting depth, note that the system program in the programmable controller can call an organization block under certain circumstances (e.g. OB3, OB13; OB2 does not influence the nesting depth).

The total nesting depth is the sum of the nesting depths of cyclic (OB1), interrupt-controlled (OB3), and time-driven (OB13) program processing. If nesting goes beyond 16 levels, the PLC goes into the ”STOP” mode with the error message ”STUEB” (block stack overflow).

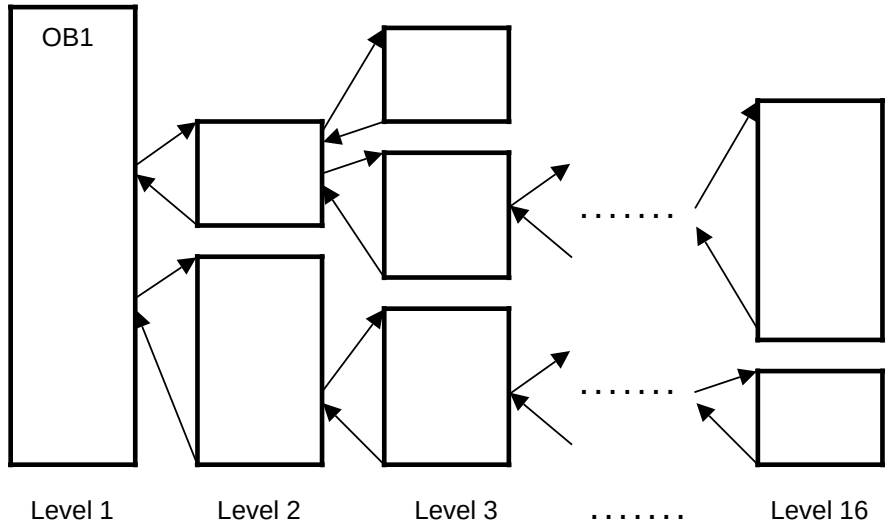


Figure 7-2. Nesting

7.3 Block Types

The following table lists the most important characteristics of the individual block types:

Table 7-2. Comparison of Block Types

	OB[1]	PB	SB	FB[2]	DB[3]
Number	255 OB1 to OB255	256 PB0 to PB255	256 SB0 to SB255	256 FB0 to FB255	256 DB0 to DB255
Length (max.)	8 KB	8 KB	8 KB	8 KB	8 KB
Operations set (contents)	Basic operations	Basic operations	Basic operations	Basic, supple-mentary, system operations	Bit pattern numbers texts
Representa-tion methods	STL, CSF, LAD	STL, CSF, LAD	STL, CSF, LAD	STL	
Block header length	5 words	5 words	5 words	5 words	5 words

1 The operating system calls up particular OBs automatically.
2 Function blocks are already integrated into the operating system (see section 9.2).
3 Data block DB0 is reserved. Data blocks DB1 and DB252, DB253 and DB254 havae special functions.

Block Structure

Each block consists of the following parts.

- Block header specifying the block type, number, and length.
  The programmer generates the block header when it transforms the block.
- Block body with the STEP 5 program or data.

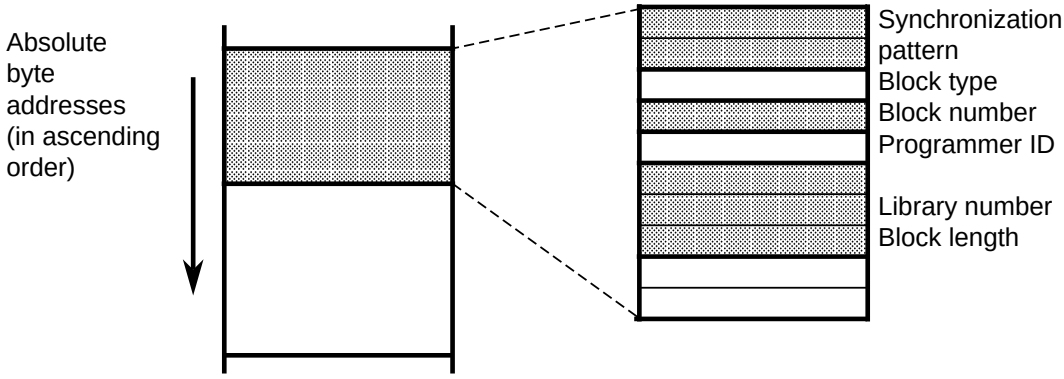


Figure 7-3. Structure of a Block Header

Programming

Program your blocks as follows (does not apply to data blocks).

1. Specify the block type (e.g., PB).
2. Specify the block number (e.g., 27).
3. Enter the control program statements.
4. Terminate the block with the ”BE” statement.

7.3.1 Organization Blocks (OBs)

Organization blocks form the interface between the operating system and the control program. Organization blocks are handled in one of the following three ways.

- One organization block (OB1) is called cyclically by the operating system.
- Some organization blocks are event-driven or time-driven.They are called by the operating system in response to events or at certain times.
  - By a switch from STOP to RUN
  - By a switch from Power OFF to Power ON (OB21 and OB22)
  - By interrupts (OB2, OB3, OB13)
  - By battery failure (OB34)
  - Error recognition (OB37)
- Some other organization blocks represent operating functions (similar to the integral function blocks). They can be called by the control program (OB31 and OB251). OBs 31 and 251 are already programmed and cannot be changed by you. You simply call these OBs in your control program.

Organization blocks OB1, 2, 3, 13, 34 and 37 are called by the S5-95F's operating system, and may not be called in your user program (S5-95F could otherwise go to STOP with a synchronization error).

Table 7-3. Overview of Organization Blocks

OB No.	Function
OB1	Cyclic program processing
Interrupt-driven program processing	
OB2	Interrupt-driven program processing (interrupts with hardware interrupt)
OB3	Interrupt-driven program processing (interrupts with software interrupt/counter overflow)
OB13	Time-controlled program processing
Handling the start-up procedures	
OB21	When starting manually (STOP to RUN)
OB22	When power returns
Handling programming errors and device errors	
OB34	Battery failure
OB37	Error handling
Special functions	
OB31	Scan time triggering (hardware watchdog)
OB251	PID control algorithm

Organization blocks OB2, OB3, OB13, OB21, OB22, OB34 and OB37 are called and processed by the operating system, even if OB1 has not been programmed.

Figure 7-4 shows how to set up a structured control program. It also illustrates the significance of organization blocks.

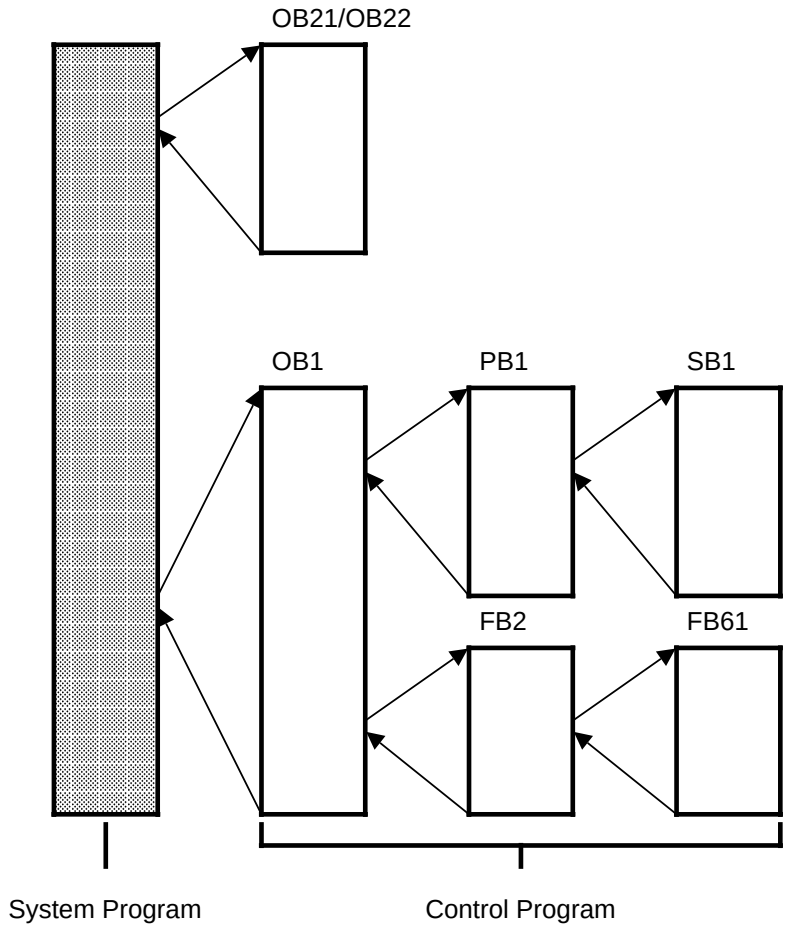


Figure 7-4. Example of Organization Block Use

7.3.2 Program Blocks (PBs)

Self-contained program parts are programmed in program blocks.

Special feature : Control functions can be represented graphically in program blocks.

Call

Block calls JU and JC activate program blocks. You can program these operations in all block types except data blocks. Block call and block end cause the RLO to be reloaded. However, the RLO can be included in the ”new” block and be evaluated there.

7.3.3 Sequence Blocks (SBs)

Sequence blocks are special program blocks that process sequence controls. They are treated like program blocks. You can use sequence blocks in the control program, but you can also use them like program blocks.

7.3.4 Function Blocks (FBs)

Frequently recurring or complex control functions are programmed in function blocks.

Function blocks have the following special features.

- FBs can be assigned parameters.
  Actual parameters can be assigned when the block is called.
- FBs have an extended set of operations not available to other blocks.
- The FB program can be written and documented in STL only.

Function blocks can be divided into the following types:

- FBs that you can program
- FBs that are integrated in the operating system (see section 9.2)
- FBs that are available as software packages (Standard Function Blocks, see Catalog ST 57)
  Standard function blocks for the S5-95F are specially protected and can neither be read nor changed. Blocks without a prototype test cannot be executed in the S5-95F.

Block Header

Besides the block header, function blocks have organizational forms that other blocks do not have.

A function block's memory requirements consist of the following.

- Block header as for other blocks (five words)
- Block name (five words)
- Block parameter for parameter assignment (three words per parameter).

Creating a Function Block

In contrast to other blocks, parameters can be assigned to FBs.
To assign parameters, you must program the following block parameter information.

- **Block Parameter Name (Formal Operand)**
  Each block parameter as a formal operand is given a designation (DES). Under this designation it is replaced by an actual operand when the function block is called.
  The name can be up to four characters long and must begin with a letter of the alphabet. You can program up to 40 parameters per function block.

- **Block Parameter Type**
  You can enter the following parameter types.
  - I input parameters
  - Q output parameters
  - D data
  - B blocks
  - T timers
  - C counters

  In graphic representation, output parameters appear to the right of the function symbol. Other parameters appear to the left.

- **Block Parameter Data Type**
  You can specify the following data types.
  - BI for operands with bit address
  - BY for operands with byte address
  - W for operands with word address
  - K for constants

When assigning parameters, enter all block parameter specifications.

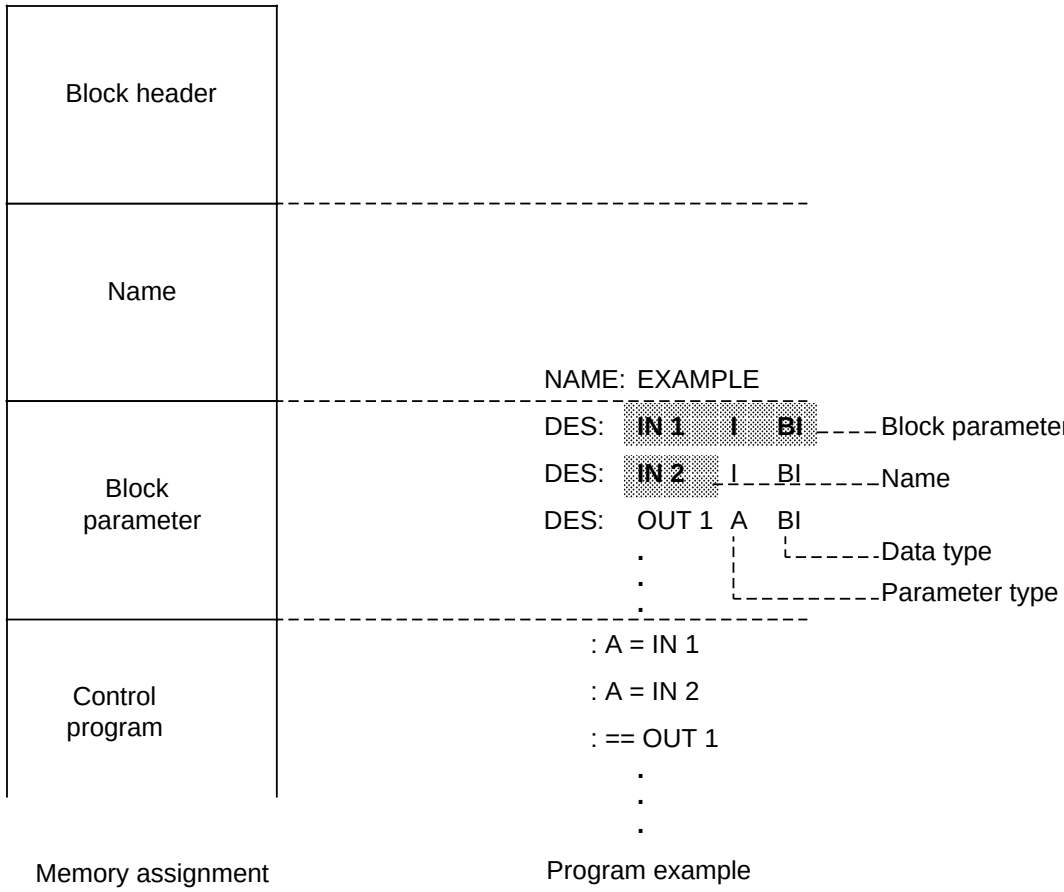


Figure 7-5. Programming a Function Block Parameter

Table 7-4. Block Parameter Types and Data Types with Permissible Actual Operands

Parameter Type	Data Type	Permissible Actual Operands
I, Q	BI for an operand with bit address	I x.y Inputs Q x.y Outputs F x.y Flags
	BY for an operand with byte address	IB x Input bytes QB x Output bytes FY x Flag bytes DL x Data bytes left DR x Data bytes right PY x Peripheral bytes*
	W for an operand with word address	IW x Input words QW x Output words FW x Flag words DW x Data words PW x Peripheral words*
D	KM for a binary pattern (16 digits) KY for two absolute numbers, one byte each, each in the range from 0 to 255 KH for a hexadecimal pattern (maximum 4 digits) KS for a character (maximum 2 alphanumeric characters) KT for a time (BCD-coded time) with time base 1.0 to 999.3 KC for a count (BCD-coded) 0 to 999 KF for a fixed-point number in the range from -32768 to +32767	Constants
B	Type designation not permitted	DB x Data blocks. The C DBx operation is executed. OB x Organization blocks are called unconditionally (JU ... x). FB x Function blocks (permissible without parameters only) are called unconditionally (JU..x). PB x Program blocks are called unconditionally (JU..x). SB x Sequence blocks are called unconditionally (JU..x).
T	Type designation not permitted	T Timer. The time should be assigned parameters as data or be programmed as a constant in the function block.
C	Type designation not permitted	C Counter. The count should be assigned parameters as data or be programmed as a constant in the function block.

* Not permitted for integral FBs

Calling a Function Block

Like other blocks, function blocks are stored under a specific number in the program memory (e.g., FB47). The numbers 240 to 255 are reserved for integrated function blocks.

You can program function block calls in all blocks except data blocks and OB2.

The call consists of the following parts.

- Call statement
  - JU FBx unconditional call (***J***ump ***U***nconditional)
  - JC FBx call if RLO = 1 (***J***ump ***C***onditional)
- Parameter list (only if block parameters were defined in the FB)

Function blocks can be called only if they have been programmed. When a function block call is being programmed, the programmer requests the parameter list for the FB automatically if block parameters have been defined in the FB.

Setting Parameters for a Function Block

The program in the function block specifies how the formal operands (parameters defined as ”DES”) are to be processed.

As soon as you have programmed a call statement (for example JU FB2), the programmer displays the ***parameter list***. The parameter list consists of the names of the parameters. Each parameter name is followed by a colon (:).You must assign actual operands to the parameters.The actual operands replace the formal operands defined in the FB when the FB is called, so that the FB ”in reality”operates with the actual operands.

A parameter list has a maximum of 40 parameters.

Example: The name (DES) of a parameter is IN1, the parameter type is I (as in input), the data type is BI (as in bit).
The formal operand for the FB has the following structure.
DES: IN1 I BI

Specify in the parameter list of calling block which actual operand is to replace the formal operand in the FB call. In our example it is : I 1.0.
Enter in the parameter list: IN1: I 1.0

When the FB is called, it replaces the formal operand ”IN1” with the actual operand ”I 1.0”.

Figure 7-6 provides you with a detailed example of how to set parameters for a function block.

The FB call takes up two words in the internal program memory. Each parameter takes up an additional memory word.

You can find the memory requirements for standard function blocks and the run times in the specifications of Catalog ST 57.

The designations (DES) of the function block inputs and outputs appearing during programming on the programmer, as well as the name of the function block, are stored in the function block itself. Therefore, you must transfer all necessary function blocks to the program diskette (for off-line programming) or directly into the program memory of the programmable controller before programming on the programmer.

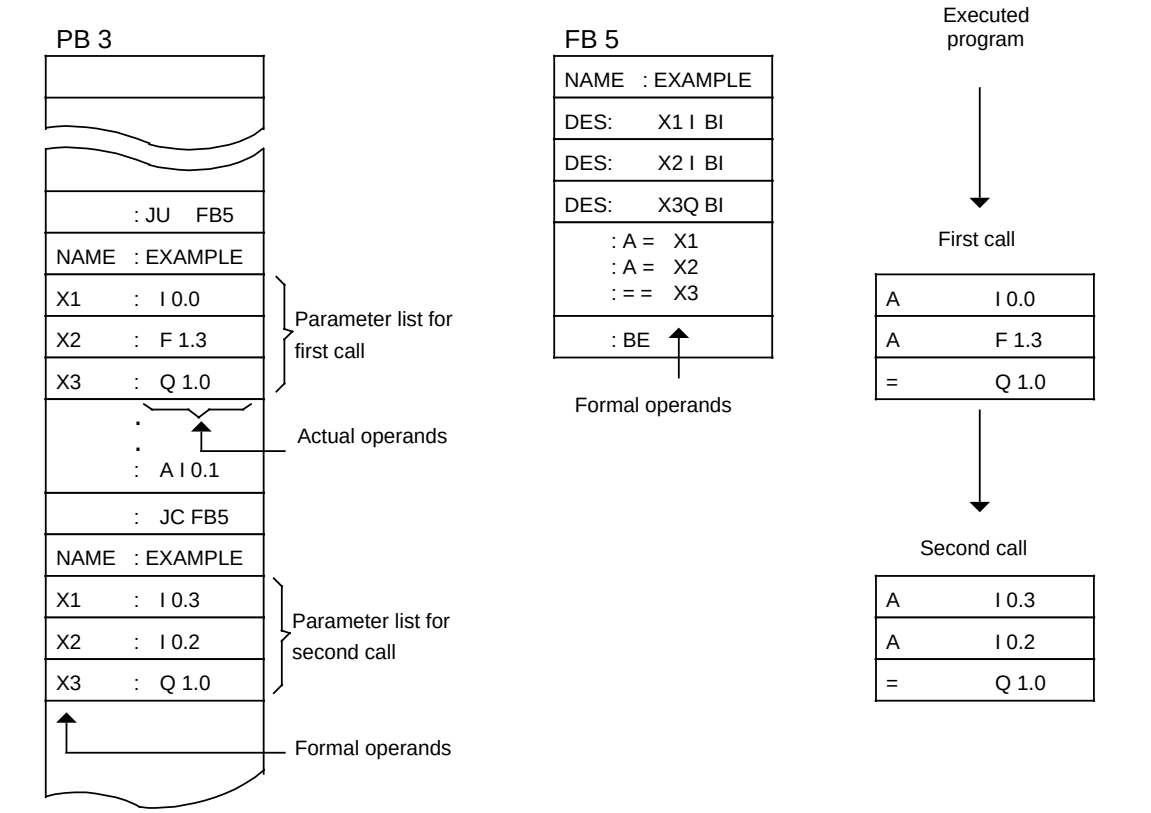


Figure 7-6. Programming a Function Block

7.3.5 Data Blocks (DBs)

Data blocks store data to be processed in a program.

The following data types are permissible.

- Bit pattern (representation of controlled system states)
- Hexadecimal, binary or decimal numbers (times, results of arithmetic operations)
- Alphanumeric characters (ASCII message texts)

Note

In the safety mode, you cannot change the contents of data blocks via the programmer. The only exception is the access to the parameter control DB in the STOP mode.
The following data blocks are provided for special S5-95F functions:

- DB1 Parameterization of the S5-95F
- DB252 SINEC L1 failsafe data path 1
- DB253 SINEC L1 failsafe data path 2
- DB254 System event DB

Programming Data Blocks

Begin programming a data block by specifying a block number. If the information takes up less than 16 bits, the high-order bits are padded with zeros. Data input begins at data word 0 and continues in ascending order. A data block can hold up to 256 data words. You can call up or change the data word contents with load or transfer operations.

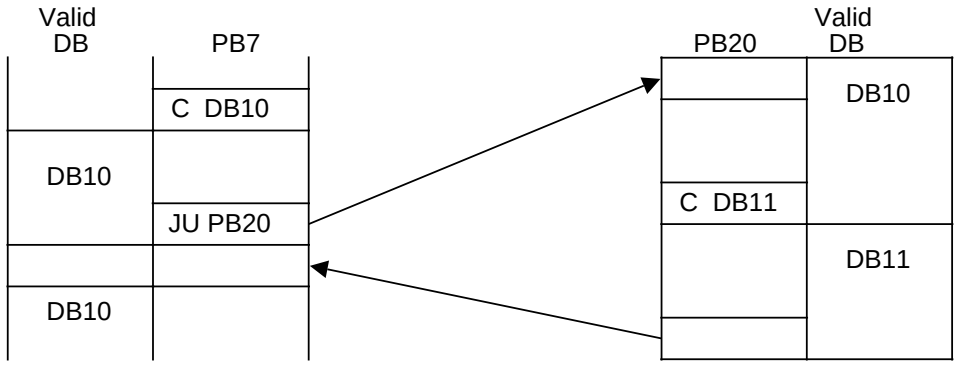
Input					Stored Values	
0000	:	KH	=	A13C	DW0	$A13C_H$
0001	:	KT	=	100.2	DW1	2100_H
0003	:	KF	=	+21874	DW2	5572_H

Figure 7-7. Data Block Contents

You can also generate or delete data blocks in the control program (see section 8.1.8).

Program Processing with Data Blocks

- A data block must be called in the program with the C DBx operation (x = DB number).
- Within a block, a data block remains valid until another data block is called.
- When the program jumps back into the higher-level block, the data block that was valid before the block call is again valid.
- After OB1, 3, 13, 21, 22, 34 and 37 have been called **by the operating system**, no DB is valid.



When PB20 is called, the valid data area is entered into a memory.
When the program jumps back, this area is reopened.

Figure 7-8. Validity Areas of Data Blocks

Function of DB1

DB1 is provided for parameterization of the S5-95F. It contains (default) values that you can either accept or change (see section 9.3). DB1 is evaluated once during start-up, in other words, either after Power ON or after a transition from STOP to RUN.

Parameter Control Data Block for Programmer Operation in the Safety Mode

In the safety mode, you must neither change the control program nor the data blocks via the programmer. The only exception is the parameter control DB. You should use the parameter control DB always when you wish to change individual values not earlier than ”during operation” via the programmer, e.g. if you wish to change recipes. In the safety mode, changing via the programmer is possible only in the STOP mode with the programmer function ”Output block”.

To ensure that your system does not go into an impermissible state, you must check all values, which you operate in the safety mode, for plausibility and reliability. This check is to be carried out in a filter program in OB21 or OB22.

You specifiy the number of parameter control DBs via COM 95F; it must not lie in the range of the DBs with constant contents.

7.4 Program Processing

Some of the organization blocks (OBs) are responsible for structuring and managing the control program.

These OBS can be grouped according to the following assignments.

- OBs for START-UP program processing
- One OB for cyclic program processing
- OBs for time-controlled program processing
- OBs for (process) interrupt-driven program processing

The S5-95F has additional OBs whose functions are similar to those of integral function blocks (e.g., PID control algorithm). These OBs are described in Chapter 9 ”Integrated Blocks”.

Section 7.3.1 summarizes all of the OBs.

7.4.1 START-UP Program Processing

In the START-UP mode, the operating system of the programmable controller automatically calls up a start-up OB if the OB has been programmed.

The following start-up OBs are available:

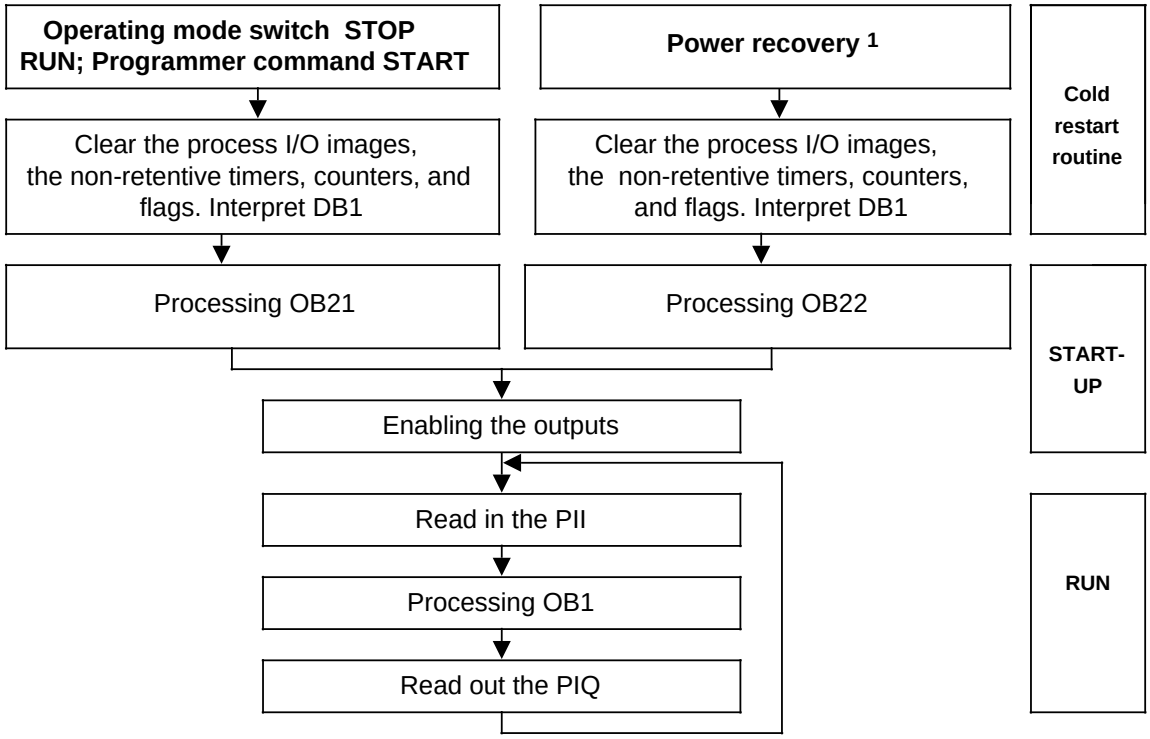
- OB21 is called up for a manual cold restart.
- OB22 is called up for an automatic cold start after power recovery if the programmable controller was previously in the RUN mode.

If you have programmed start-up OBs, then they are processed before the cyclic program processing starts. The start-up OB program is appropriate, for example, for a one-time presetting of certain system data.

If you access the PII in OB21/22 or if you directly access the onboard I/Os, then you always receive the value ”0”. If you write to the PIQ or to the onboard I/Os via direct access in the OB21/22, then your change is stored and becomes effective only after OB1 has been processed.

Features of the start-up blocks (OB21, OB22):

- The red and green LEDs light up.
- Timers TO to T63 are processed, timers T64 to T127 are not processed
- Scan monitoring is not activated.
- Interrupts are not being processed.



1 This is the procedure if the programmable controller was in the "RUN" mode when the power went off, if the mode switch was still on RUN when the power was restored, and if the battery was inserted.

Figure 7-9. Setting the Start-Up Procedure

The following two examples show you how you can program a start-up OB.

Example: Programming OB21/OB22

Example	STL	Explanation
After the operating mode switch causes a cold restart, flag bytes 0 to 9 are preset with ”0”. The other flag bytes are retained since they contain important machine functions.	L KH 0 T FW 0 T FW 2 T FW 4 T FW 6 T FW 8 BE	Value ”0” is loaded in ACCU 1and transferred into flag words 0, 2, 4, 6, and 8.

7.4.2 Cyclical Program Processing

The operating system calls OB1 cyclically. If you want to have structured programming, you should program only jump operations (block calls) in OB1. The blocks you call up, PBs, FBs, and SBs, should contain completed functional units in order to provide a clearer overview.

The S5-95F monitors the duration of the cyclical program processing. At the beginning of the program cycle it triggers two independent cycle time monitors (watchdog).
If the scan cycle time trigger is not reset within the scan monitoring time, the programmable controller automatically goes to the "STOP" mode and disables the output modules. The monitoring time is exceeded, for example, if you program an infinite loop or if a malfunction has occurred in the programmable controller.

The monitoring time of the first cycle time monitor is defined via the S5-95F hardware to be approx. 700 ms. If your control program is so complex that it can not be processed within 700 ms, you can use OB 31 to lengthen (retrigger) the monitoring time in the control program .

The monitoring time of the second cycle time monitor is checked by the operating system. The monitoring time is set in DB1 and you cannot retrigger it in the control program. For the second cycle time monitor (software watchdog) you can set monitoring times up to 2550 ms.

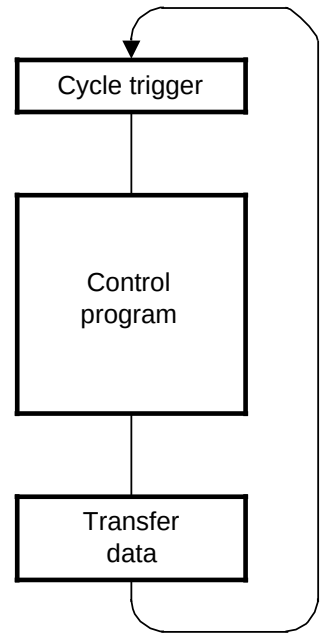


Figure 7-10. Cyclical Program Processing

Extension of the Cycle Time through Interrupts

The following paragraphs are intended for the system expert and explain the correlation between interrupts and the resulting extension of the cycle time.

1. Extension of Cycle Time

An S5-95F cycle consists of the operating system execution time and the execution time of the control program (OB1).

The cycle is interrupted by several events, such as the internal updating of the timers. Execution of these interrupts represents an additional load for the system. The load (proportion of execution time in %) is in the following called quota Q. Through the quota Q, the S5-95F cycle is extended by a certain factor.

The following applies for the factor of extension:

$$\text{Extension } D = \frac{1}{1 - \text{sum of all quotas } Q} \qquad \text{where } D = \frac{T_{PLC} \text{ with interrupts}}{T_{PLC} \text{ without interrupts}}$$

2. Cycle Stability

The formula shows that the extension D is a non-linear function. An important criterion is therefore what additional interrupt load will be added to an existing interrupt load. Typical quotas for loading through interrupts are specified in the table below.

Example: The following example shows how loading through interrupts affects the cycle.

1st case: A system with no load with a cycle time of, for example, 100 ms is extended through interrupts with a quota of 25 % by a factor of D = 1.333 (the cycle time is extended by 33 % to 133 ms).

2nd case: A system which is already loaded with a quota of 50 % and which has, for example, a cycle time of 200 ms is loaded with addtional interrupts by an additional quota of 25 %. The cycle time already extended by a factor of D = 2 is extended by the factor of D = 4 to 400 ms.

Rule: Limit the **system load through additional interrupts to max. 50 %**, so that sporadic peaks cannot result in overproportional increases in the PLC cycle time.

If the cycle time extension exceeds the factor D = 2, the S5-95F may fail to perform an automatic warm restart following a power outage.

3. Typical Execution Time Quotas of Additional Interrupts

Interrupts	Execution Time Quota Q
1. Programmer, depending on operation	5-10 %
2. SINEC L1	max. 8 % with one LAN max. 12 % with two LANs Quota depends on: • SINEC L1 polling list • Frame length • Safety time • Message frequency
3. OB2	$Q = \frac{0.4\ ms + T_{OB2}/ms}{\text{average OB2 interval/ms}}$
4. OB3 (triggered by interrupt DI)	$Q = \frac{0.4\ ms + T_{OB3}/ms}{\text{average OB3 interval/ms}}$
5. OB3 (triggered by counter)	$Q = \frac{0.1\ ms + T_{OB3}/ms}{\text{average triggering time/ms}}$
6. OB13 without direct access	$Q = \frac{1 \text{ to } 5\ ms + T_{OB13}/ms}{\text{OB13 interval time/ms}}$
7. OB13 with direct access	$Q = \frac{2 \text{ to } 10\ ms + T_{OB13}/ms}{\text{OB13 interval time/ms}}$

7.4.3 Maximum Response Time with Cyclical Program Processing

The reaction to a change in the input signal is a change in the output signal. The time between the change in the input signal and the change in the output signal is called response time.

To determine the response time, you must know the following times:

- Program processing time
- Operating system runtime, including the read and write cycles of the onboard I/Os (25 ms)*
- Processing time of the external I/Os (2 ms per bus module)
- Delay time of the input modules (5 ms)

Calculating the Maximum Response Time for the Onboard I/Os

Figure 7-11 shows a schematic of the S5-95F program processing.

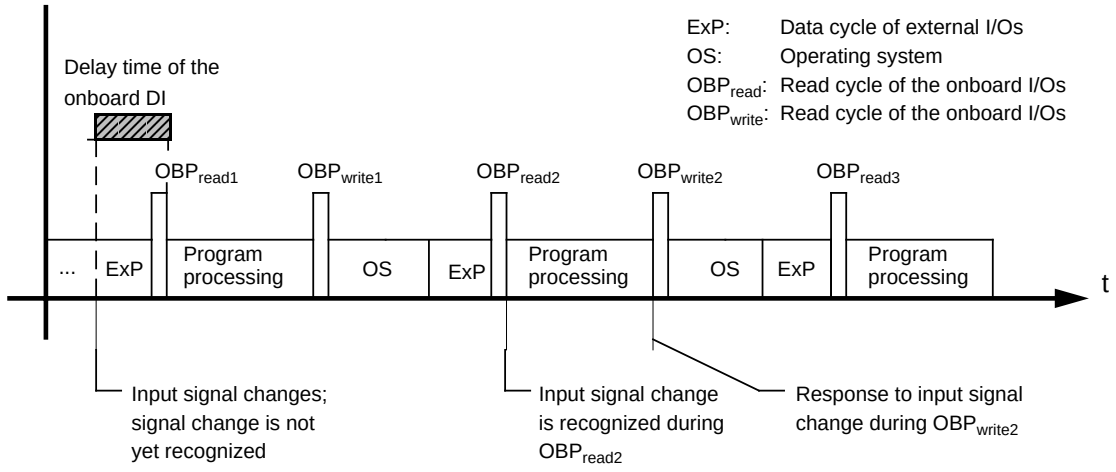


Figure 7-11. Response Time of the Onboard I/Os (Worst Case)

You calculate the maximum response time $T_{OB1\ response,\ OBP}$ with cyclical program processing via the onboard I/Os as follows:

$T_{OB1\ response,\ OBP}$ = 2 x program processing time
+1 x operating system runtime (25 ms)*
+ 1 x processing of external I/Os (2 ms per bus module)
+1 x delay time of the input modules (5 ms)

Expressed in OB1 cycles, the following applies:

$T_{OB1\ response,\ OBP}$ = 2 × OB1 cycle time
- 1 x operating system runtime (25 ms)*
- 1 x processing of external I/Os (2 ms per bus module)
+ 1 x delay time of the input modules (5 ms)

= 2 x OB1 cycle time - 25 ms* + 5 ms - number of bus modules x 2 ms
= 2 x OB1 cycle time - 20 ms - number of bus modules x 2 ms

The integrated cycle time statistics (see section 15.7) are an aid enabling you to initially assess the cycle time.

* Times increase in the case of loading through interrupts or programmer operation (see section 7.4.2).

Calculating the Maximum Response Time of the External I/Os

Figure 7-12 shows a schematic of the S5-95F program processing.

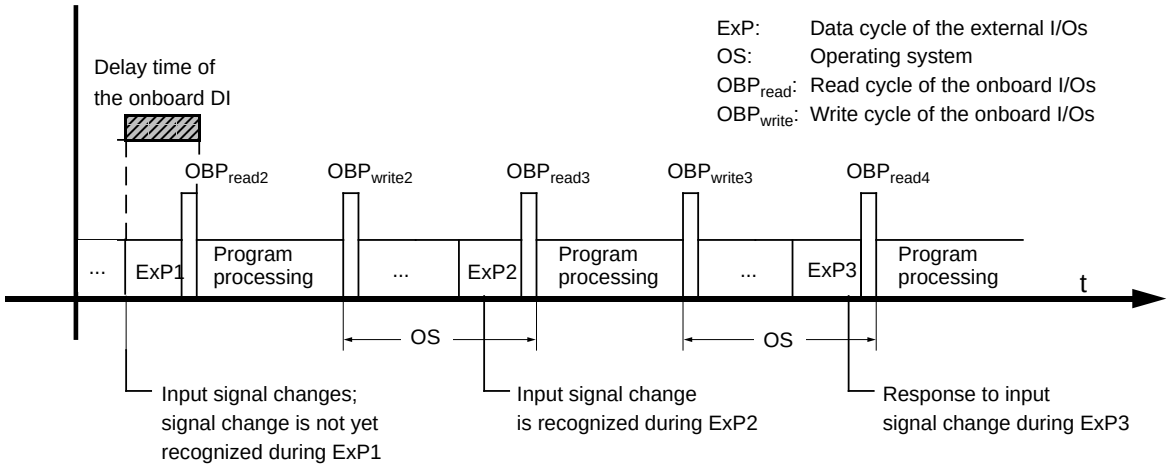


Figure 7-12. Response Time of the External I/Os (Worst Case)

You calculate the maximum response time $T_{OB1\ response}$ with cyclical program processing via the external I/Os as follows:

$T_{OB1\ response,\ ExP}$ = 2 x program processing time
+ 2 x operating system runtime (25 ms)*
+ 1 x data cycle time of the external I/Os (max. 7 ms)
+ 1 x delay time of the input modules (5 ms)

Expressed in OB1 cycles, the following applies:

$T_{OB1\ response,\ ExP}$ = 2 x OB1 cycle time
+ 1 x data cycle time of the external I/Os (max. 7 ms)
+ 1 x delay time of the input modules (5 ms)

The data cycle time given in the formula depends on the length of the shift register for all modules. A time of 25 µs is required for each data bit. In a fully configured S5-95F system, the data cycle time is a maximum of 7 ms (see section 2.4.2).

The integrated cycle time statistics (see section 15.7) are an aid enabling you to initially assess the cycle time

Extension of the Response Time through STATUS Programmer Function

> **Note**
>
> During the STATUS programmer function, the S5-95F processes the blocks in a special mode; the runtime of the blocks increases considerably and is command-dependent. Please note that this also results in an increase in the response time. Therefore program your blocks as short as possible.

* Times increase in the case of loading through interrupts or programmer operation (see section 7.4.2).

7.4.4 Time-Controlled Program Processing

Time-controlled program processing can be defined as a (periodic) time signal causing the CPU to interrupt cyclic program processing and to process a specific program. Once this program has been processed, the CPU returns to the point of interruption in the cyclic program and continues processing.

Prerequisites for Time-Controlled Program Processing

Time-controlled program processing is possible only if the following prerequisites have been fulfilled:

- The OB13 call-up interval must be programmed larger than 0 ms with COM 95F.
- Organization block OB13 must be programmed.
- The S5-95F must be in the RUN mode.
- Interrupt processing must not be disabled by the IA (disable interrupt) operation (see section 8.2.8).

Cyclical program processing continues if OB13 is not programmed.

Periodic Program Processing in OB13

If you use COM95F to program a call-up interval for OB13, then the S5-95F calls OB13 periodically. The instant for the actual processing of OB13 depends on the following:

- The call-up interval programmed
- The delay t, which depends on the instant of the next synchronization and on the operating system functions, which are processed in the synchronization routine.

The delay t is a statistically fluctuating variable. The typical range of fluctuation lies between 0 and 10 ms. Reference values for the maximum OB13 processing delay are given in Table 7-5.

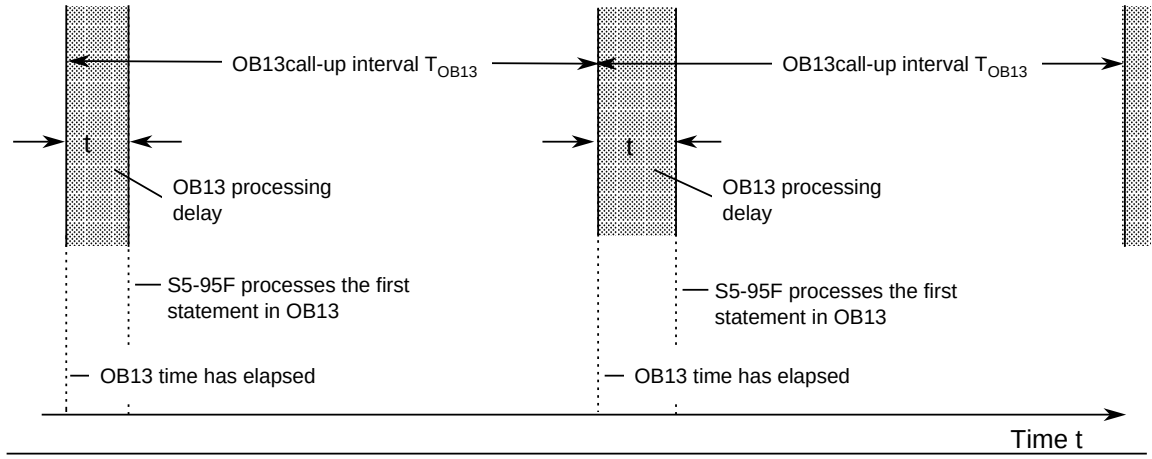


Figure 7-13. Call-Up Interval and Delay of OB13 Processing

In the ideal case, OB13 is processed directly when the previous OB13 time has elapsed (t = 0). Figure 7-13 shows that the interval between two OB13 processings can be between T_{OB13} - t_{max} and T_{OB13} + t_{max}.

Selection of the OB13 Call-Up Interval

You set the OB13 call-up interval when parameterizing the system with COM 95F. The OB13 call-up interval must fulfill the following conditions:

OB13 call-up interval > OB13 runtime + maximum delay t of OB13 processing

If you have programmed an OB13 call-up interval that does not fulfill the above equation, this results in a time interrupt error. In this case, the S5-95F goes into STOP.

Reference Values for OB13 Processing Delays

The OB13 processing delay depends on the loading of the system. Table 7-5 gives reference values for an S5-95F programmable controller with a low system load. The delay of OB13 processing depends on the following:

- Use of the OB2 and/or OB3 interrupts
- Direct accesses to external I/Os in OB13
- Configuration of external I/O modules
- Use of the SINEC L1 data interchange facilities
- Programmer operation

Table 7-5. Reference Values for the OB13 Call-Up Delay (No Use of SINEC L1)

OB2 and OB3 Interrupt Processing	Direct Access to External I/Os in OB13	Configuration with External I/Os (per Subunit)	OB13 Call-Up Delay
no	no	no external I/Os	10 ms
no	yes	small (approx. 40 data bits)	20 ms
no	yes	large (approx. 300 data bits)	30 ms
yes	no	no external I/Os	20 ms
yes	yes	small (approx. 40 data bits)	30 ms
yes	yes	large (approx. 300 data bits)	40 ms

If you use the SINEC L1 LAN you must add the following times to the values given in Table 7-5:

- Basic load for SINEC L1 . . . . . 10 ms
  (independent of whether configuration consists of one or two buses)
- For each bus you use to send safe data . . . . . 2.5 ms
- For each bus you use to receive safe data . . . . . 2.5 ms

OB13 Call Interval and Module Replacement in RUN Mode

If a module is replaced in RUN mode, the S5-95F initializes the I/O bus, which can take up to 300 ms.
If you want to replace a module in RUN mode, the OB13 call interval must therefore be at least 350 ms.

Interrupt Possiblities

OB13 can interrupt the cyclic program after any STEP 5 statement. OB13 itself can be interrupted by process interrupts (OB2 and OB3). If OB13 has been interrupted for interrupt processing, the time-controlled program processing in OB13 is terminated afterwards.

OB13 can ***not*** interrupt the following:

- Processing of the data cycles
- Interrupt-driven program processing (OB2, OB3)
- Current time-controlled program processing (OB13)
- SINEC L1 processing.

Disabling/Enabling the Call-Up

Use the IA command in the control program to disable the OB13 call-up and the RA command to enable it. A call-up request can be stored during a call-up disable. The default is RA.

Please refer also to section 8.2.8 "Disabling/Enabling Interrupt".

Saving Data

If you use flags in the time-controlled program that you also use in the cyclic program, then you must save these in a data block during time OB processing. After time OB processing you must restore them into the flag area.

Note

When processing OB13, you may not exceed the block nesting depth of 16 levels.

Reading Out the Interrupt PII

When OB13 is called, the signals of the external input modules are read into the interrupt PII. The interrupt PII can be scanned in OB13 by means of the L PY x or L PW x load operations (load byte x or word x of the interrupt PII in ACCU 1). There is an interrupt input data cycle prior to time-driven program processing. The response time of the cyclical program processing is prolonged by the interrupt data cycle time.

Writing to the Interrupt PIQ

Data to the external I/Os can be written to the interrupt PIQ by means of transfer operations T PY x or T PW x. At the same time, data is written to the "normal" PIQ.

After termination of OB13, the data that has been transferred to the interrupt PIQ is output to the I/O modules in an interrupt output data cycle (prior to "normal" program processing). The response time of the cyclical program processing is prolonged by the interrupt data cycle time.

Note

The interrupt output data cycle is executed only if data has been written to the PIQ.

Time Interrupt Error

If the system is so heavily loaded that it cannot process OB13 during the defined OB13 call-up interval, this leads to a time interrupt error (OB13 is called again before OB13 processing is completed). In the case of a time interrupt error, the S5-95F goes to STOP. The same error can occur if OB13 is frequently interrupted by interrupt calls.

Remedy:

- Extend your OB13 call-up interval
- Reduce the program in OB13
- Minimize the number of interrupt calls

7.4.5 Maximum Response Time with Time-Controlled Program Processing

Response time is the time between the change in an input signal and the change in an output signal.

The maximum response time with time-controlled program processing depends on the following:

- On the OB13 call-up interval set
- On the delay time t (see section 7.4.4)
- On the execution time T_{OB13} of OB13
- On the delay time of the input module
- On the data cycle (if external I/O modules are used)

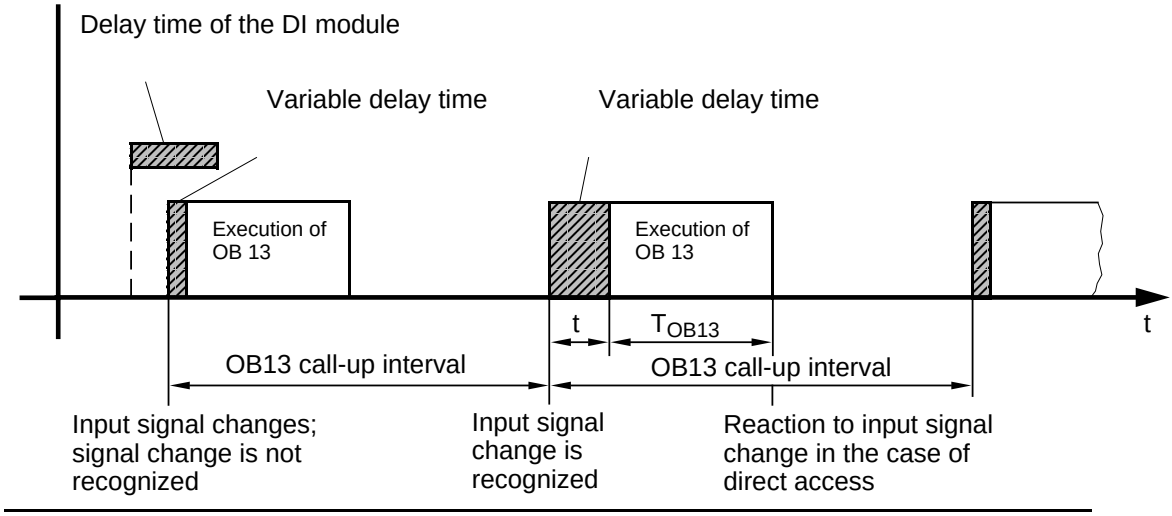


Figure 7-14. Response Time with Time-Controlled Program Processing

Under worst-case conditions, the following applies for the response time with time-controlled program processing and programmed direct access:

$T_{OB13\ response}$
- 1 x OB13 call-up interval
- \+ OB13 processing delay (max. one OB13 call-up interval)
- \+ OB13 execution time plus 1 data cycle if the S5-95F is configured with external I/Os and 1 data cycle if direct access to the external I/Os has been programmed in OB13 (max. one OB13 call-up interval)
- \+ Delay time of the input module

With direct access to the onboard I/Os you will obtain the shortest response time. If you have programmed neither OB2/OB3 interrupts nor direct access to external I/Os, then the response time for onboard I/Os with direct access is as follows:

$T_{OB13\ response}$ 20 ms + OB13 call-up interval + OB13 execution time.

7.4.6 Interrupt-Driven Program Processing

Interrupt-driven program processing is initiated when a signal from the process causes the CPU to interrupt cyclic or time-controlled program processing and to execute a specific program. When this program has been scanned, the CPU returns to the point of interruption in the cyclic or time-controlled program and continues scanning at that point. For detailed information on interrupt processing please refer to Chapter 12.

7.5 Processing Blocks

Earlier sections in this chapter described how to use blocks. Chapter 8 introduces all of the operations required to work with blocks. Blocks that you have already programmed can only be changed in the test mode. The individual ways to change blocks are only briefly described. The operating instructions of the programmers used explain the necessary steps in detail.

7.5.1 Changing Programs

Program changes are not possible in the safety mode.
In the test mode, program changes can be executed with the following programmer functions independent of the type of block.

- INPUT
- OUTPUT
- STATUS

These three programmer functions make it possible for you to make the following types of changes:

- Delete, insert, or overwrite statements.
- Insert or delete segments.

7.5.2 Changing Blocks

Program changes refer to changing the contents of a block. You can also delete or overwrite a complete block. When you delete a block, it is not deleted from the program memory but simply becomes invalid. You cannot enter new information in the memory location of an invalid block. This may cause a new block not to be accepted. If a new block is not accepted, then the PG transmits the ”no space available” error message. You can delete this message by compressing the programmable controller memory. Compressing is not possible in the RUN mode of safety operation.

7.5.3 Compressing the Program Memory

Figure 7-15 illustrates what takes place in the program memory during a COMPRESS operation. Internally, one block is shifted per cycle.

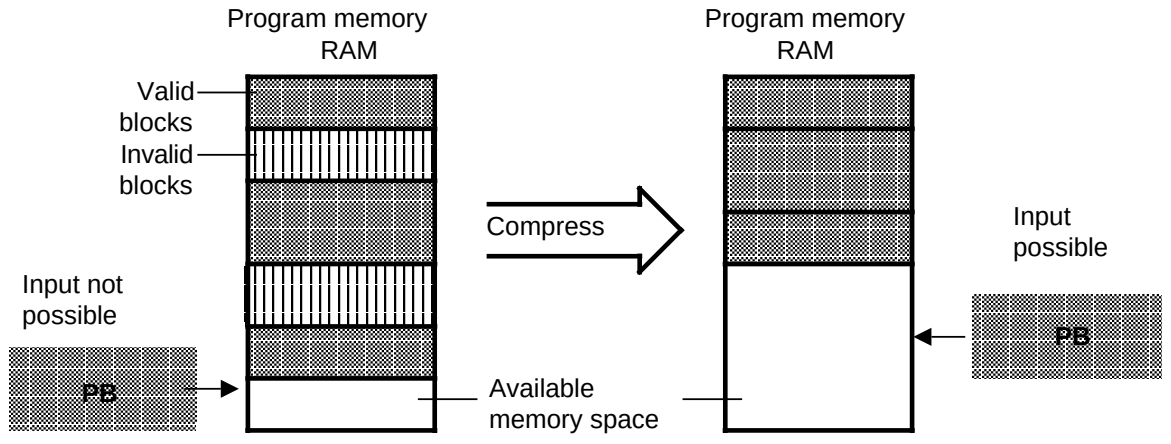


Figure 7-15. Compressing the Program Memory

You can use the COMPRESS programmer function to clean up program memory.

If there is a power failure during the compress operation when a block is being shifted and block shifting can not be completed, the programmable controller remains in the STOP mode. The ”NINEU” error message appears. Both the ”BSTSCH” and the ”SCHTAE” bits are set in the ISTACK.

Remedy: Overall reset.

7.6 Number Representation

With STEP 5 you can work with numbers in the following five representations.

- Decimal numbers from -32768 to +32767 (KF)
- Hexadecimal numbers from 0000 to FFFF (KH)
- BCD-coded numbers (4 tetrads) from 0000 to 9999
- Bit patterns (KM)
- Constant byte (two-byte representation) from 0 to 255 for each byte (KY)

Number Formats

The programmable controller is designed to process binary signal states (”0” and ”1”). Therefore programmable controllers represent all numbers internally as 16-bit binary numbers or as bit patterns.

Four bits can be combined into a tetrad (BCD) to shorten the binary code representation. The value of these tetrads can be displayed in hexadecimal representation.

Example: 16-bit binary coded number and shortened hexadecimal representation

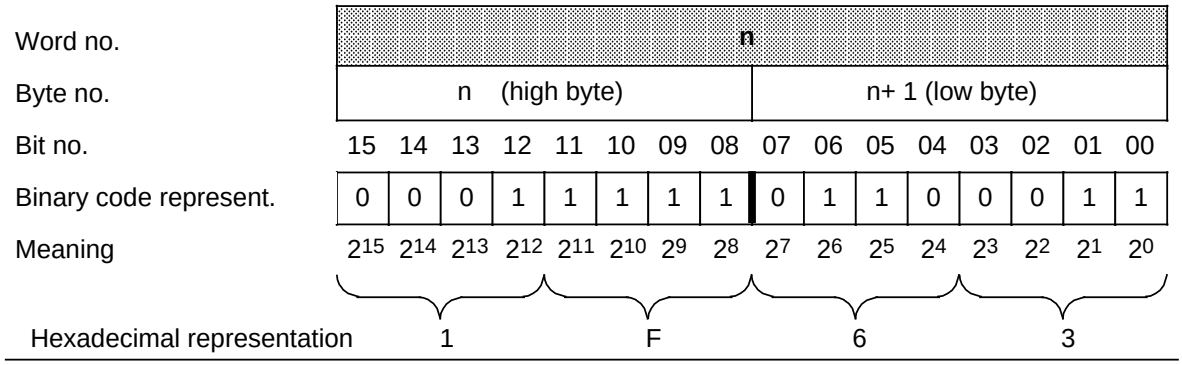


Figure 7-16. Bit Assignment of a 16-Bit Fixed-Point Binary Number

You can work with binary-coded decimals to program timers and counters in the decimal system.

BCD tetrads are defined in the range of 0 to 9.

Example: 12-bit timer or counter value in BCD and decimal formats

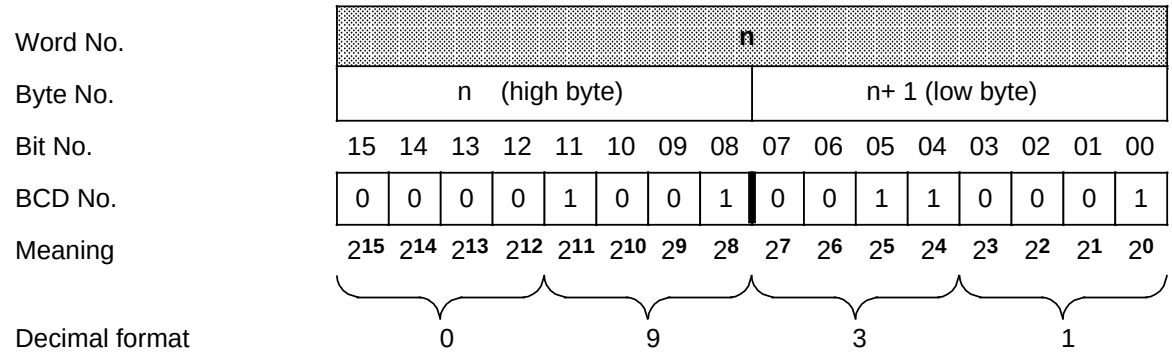


Figure 7-17. BCD and Decimal Formats

Table 7-6. Comparison of Number Formats

Binary	Decimal	BCD	Hexadecimal
0000	0	0000 0000	0
0001	1	0000 0001	1
0010	2	0000 0010	2
0011	3	0000 0011	3
0100	4	0000 0100	4
0101	5	0000 0101	5
0110	6	0000 0110	6
0111	7	0000 0111	7
1000	8	0000 1000	8
1001	9	0000 1001	9
1010	10	0001 0000	A
1011	11	0001 0001	B
1100	12	0001 0010	C
1101	13	0001 0011	D
1110	14	0001 0100	E
1111	15	0001 0101	F

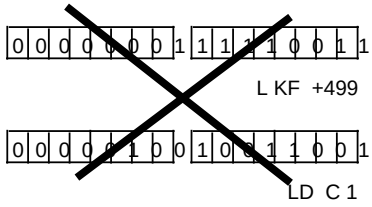
You can use the ”LD” operation to load a binary number as a BCD number for timer and counter values.

Example: Comparing a count in counter 1 with decimal number 499.
The comparison value must be stored in the accumulator by means of the load operation. In order not to have to convert the value 499 into other numerical systems (binary or hexadecimal) for input, use the ”L KF +499” statement. The number $1F3_H$ is then stored in the accumulator.

The current count must also be loaded into the accumulator.

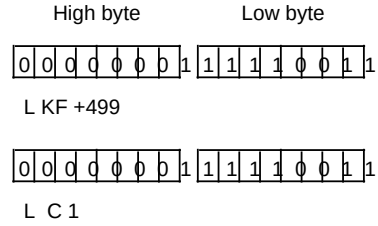
Incorrect method:

If you use the ”LD C1” statement for this purpose, the current count will be loaded in BCD. The ”!=F” comparison operation results in a not-equal-to condition since the comparison involves different formats.



Correct method:

If the ”L C1” statement is entered, the formats are identical.



8 STEP 5 Operations

8.1	Basic Operations	8 - 1
8.1.1	Boolean Logic Operations	8 - 2
8.1.2	Set/Reset Operations	8 - 7
8.1.3	Load and Transfer Operations	8 - 10
8.1.4	Timer Operations	8 - 15
8.1.5	Counter Operations	8 - 25
8.1.6	Comparison Operations	8 - 29
8.1.7	Arithmetic Operations	8 - 30
8.1.8	Block Call Operations	8 - 32
8.1.9	Other Operations	8 - 37
8.2	Supplementary Operations	8 - 38
8.2.1	Load Operation	8 - 39
8.2.2	Enable Operation	8 - 40
8.2.3	Bit Test Operations	8 - 41
8.2.4	Digital Logic Operations	8 - 43
8.2.5	Shift Operations	8 - 47
8.2.6	Conversion Operations	8 - 49
8.2.7	Decrement/Increment	8 - 51
8.2.8	Disabling/Enabling Interrupts	8 - 52
8.2.9	"DO" Operation	8 - 53
8.2.10	Jump Operations	8 - 55
8.2.11	Substitution Operations	8 - 57
8.3	System Operations	8 - 63
8.3.1	Set Operations	8 - 63
8.3.2	Load and Transfer Operations	8 - 64
8.3.3	Arithmetic Operations	8 - 66
8.3.4	Other Operations	8 - 67
8.4	Condition Code Generation	8 - 68
8.5	Sample Programs	8 - 70
8.5.1	Momentary-Contact Relay/Edge Evaluation	8 - 70
8.5.2	Binary Scaler/Binary Divider	8 - 70
8.5.3	Clock/Clock-Pulse Generator	8 - 72

Figures

8-1	Accumulator Structure	8 - 10
8-2	Execution of the Load Operation	8 - 12
8-3	Transferring a Byte	8 - 12
8-4	Output of the Current Time (Example)	8 - 18
8-5	Output of the Current Counter Status (Example)	8 - 26
8-6	Executing a "DO" Operation	8 - 54

Tables

8-1	Overview of Boolean Logic Operations	8 - 2
8-2	Overview of the Set/Reset Operations	8 - 7
8-3	Overview of Load and Transfer Operations	8 - 11
8-4	Overview of Timer Operations	8 - 15
8-5	Overview of Counter Operations	8 - 25
8-6	Overview of Comparison Operations	8 - 29
8-7	Overview of Arithmetic Operations	8 - 30
8-8	Overview of Block Call Operations	8 - 32
8-9	Overview of Other Operations	8 - 37
8-10	Load Operation	8 - 39
8-11	Enable Operation	8 - 40
8-12	Overview of Bit Operations	8 - 41
8-13	Effect of "TB" and "TBN" on the RLO	8 - 41
8-14	Overview of Digital Logic Operations	8 - 43
8-15	Overview of Shift Operations	8 - 47
8-16	Overview of Conversion Operations	8 - 49
8-17	Decrement/Increment Operations	8 - 51
8-18	Disable/Enable Interrupt Operations	8 - 52
8-19	Overview of the "DO" Operation	8 - 53
8-20	Overview of Jump Operations	8 - 55
8-21	Overview of Binary Logic Operations	8 - 57
8-22	Overview of Set/Reset Operations	8 - 58
8-23	Overview of Load and Transfer Operations	8 - 59
8-24	Overview of Timer and Counter Operations	8 - 60
8-25	"DO" Operation	8 - 62
8-26	Overview of Set Operations	8 - 63
8-27	Overview of Load and Transfer Operations	8 - 64
8-28	Arithmetic Operation	8 - 66
8-29	The "TAK" and "STS" Operations	8 - 67
8-30	Condition Code Settings for Comparison Operations	8 - 68
8-31	Condition Code Settings for Fixed-Point Arithmetic Operations	8 - 68
8-32	Condition Code Settings for Digital Logic Operations	8 - 69
8-33	Condition Code Settings for Shift Operations	8 - 69
8-34	Condition Code Settings for Conversion Operations	8 - 69

8 STEP 5 Operations

The STEP 5 programming language has the following three operation types:

- Basic Operations include functions that can be executed in organization, program, sequence, and function blocks.
- Supplementary Operations include complex functions such as substitution statements, test functions, and shift and conversion operations.

  All supplementary operations are programmed in the STL method of representation. As of version V 6.0 of the STEP 5 basic package, the CSF type of representation can also be used for some of these operations.
- System Operations access the operating system directly. Only an experienced programmer should use them.
- System operations can be input and output in STL form only.
- If you are familiar with S5-95U programming, then please note that the operations can be restricted with the S5-95F and/or can be executed differently (see section 16.2.2).

8.1 Basic Operations

Sections 8.1.1 through 8.1.9 use examples to describe the basic operations.

8.1.1 Boolean Logic Operations

Table 8-1 provides an overview of Boolean logic operations. Examples follow the table.

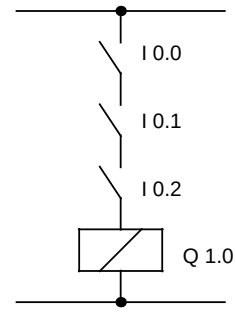
Table 8-1. Overview of Boolean Logic Operations

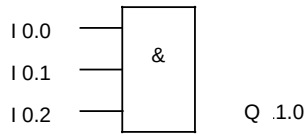
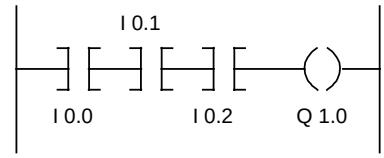
Operation	Operand		Meaning
O			**Combine AND operations through logic OR** Combine the result of the next AND logic operation (RLO) with the previous RLO through logic OR.
A(			**Combine expression encl. in parentheses through logic AND** Combine the RLO of the expression enclosed in parentheses with the previous RLO through logic AND.
O(			**Combine expression enclosed in parentheses through logic OR** Combine the RLO of the expression enclosed in parentheses with the previous RLO through logic OR.
)			**Close parenthesis** Conclude the expression enclosed in parentheses.
A			**Scan operand for "1" and combine with RLO through logic AND** The result is "1" when the operand in question carries signal state "1". Otherwise the scan results in "0". Combine this result with the RLO in the processor through logic AND[1].
O			**Scan operand for "1" and combine with RLO through logic OR** The result is "1" when the operand in question has signal state "1". Otherwise the scan results in "0". Combine this result with the RLO in the processor through logic OR[1].
AN			**Scan operand for "0" and combine with RLO through logic AND** The result is "1" when the operand in question has signal state "0". Otherwise the scan results in "0". Combine this result with the RLO in the processor through logic AND[1].
ON			**Scan operand for "0" and combine with RLO through logic OR** The result is "1" when the operand in question has signal state "0". Otherwise the scan results in "0". Combine this result with the RLO in the processor through logic OR[1].
	ID		**Parameter**
	I		0.0 to 127.7
	Q		0.0 to 127.7
	F		0.0 to 255.7
	T		0 to 127
	C		0 to 127

1 If the scan follows directly after an RLO-reloading operation (first scan), the scan result is reloaded as new RLO.

AND Operation

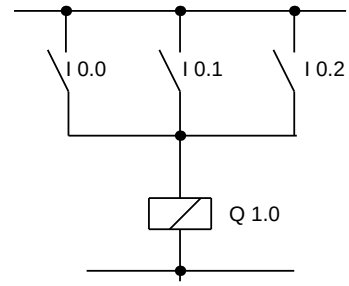
The AND operation scans to see if various conditions are satisfied simultaneously.

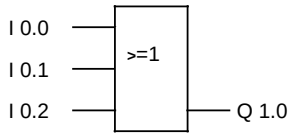
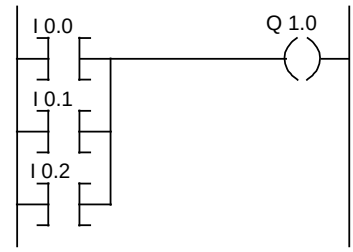
Example	Circuit Diagram
Output Q 1.0 is "1" when all three inputs are "1". The output is "0" if at least one input is "0". The number of scans and the sequence of the logic statements are at random	

STL	CSF	LAD
A I 0.0 A I 0.1 A I 0.2 = Q 1.0		

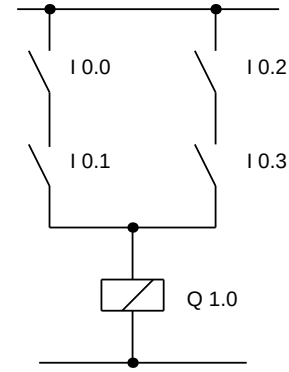
OR Operation

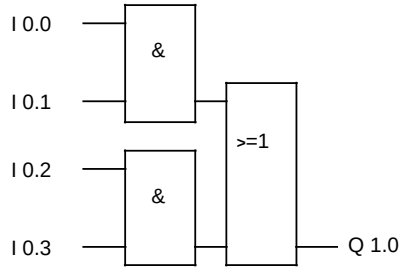
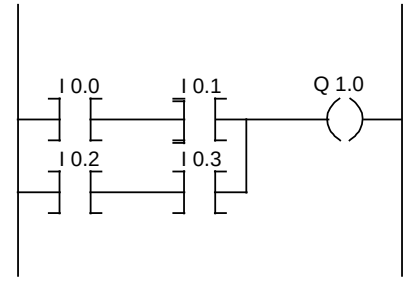
The OR operation scans to see if one of two (or more) conditions has been satisfied.

Example	Circuit Diagram
Output Q 1.0 is "1" when at least one of the inputs is "1". Output Q 1.0 is "0" when all inputs are "0" simultaneously. The number of scans and the sequence of their programming are optional.	

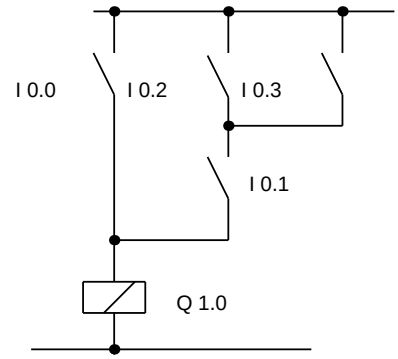
STL	CSF	LAD
O I 0.0 O I 0.1 O I 0.2 = Q 1.0		

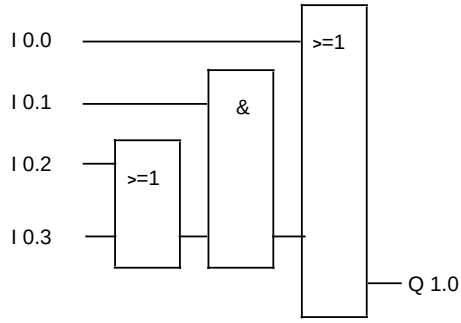
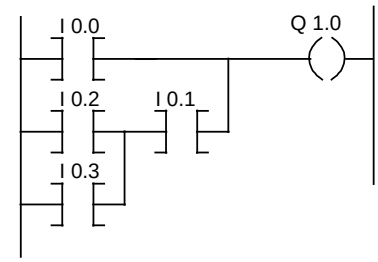
AND before OR Operation

Example	Circuit Diagram
Output Q 1.0 is "1" when at least one AND condition has been satisfied. Output Q 1.0 is "0" when neither of the two AND conditions has been satisfied.	

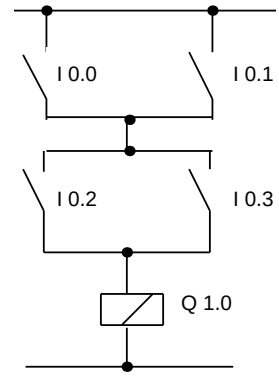
STL	CSF	LAD
A I 0.0 A I 0.1 O A I 0.2 A I 0.3 = Q 1.0		

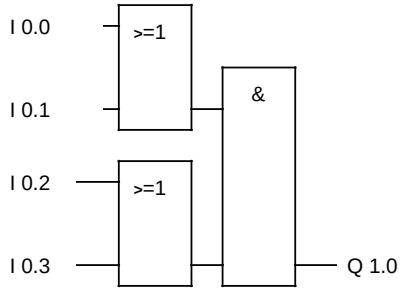
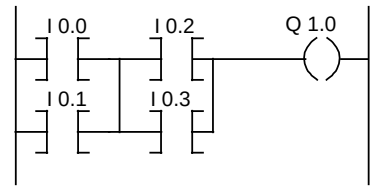
OR before AND Operation

Example	Circuit Diagram
Output Q 1.0 is "1" when one of the following conditions has been satisfied: • Input I 0.0 is "1". • Input I 0.1 and either input I 0.2 or I 0.3 is "1". Output Q 1.0 is "0" when none of the OR conditions has been satisfied.	

STL	CSF	LAD
O I 0.0 O A I 0.1 A(O I 0.2 O I 0.3) = Q 1.0		

OR before AND Operation

Example	Circuit Diagram
Output Q 1.0 is "1" when both OR conditions have been satisfied. Output Q 1.0 is "0" when at least one OR condition has not been satisfied.	

STL	CSF	LAD
A(O I 0.0 O I 0.1) A(O I 0.2 O I 0.3) = Q 1.0		

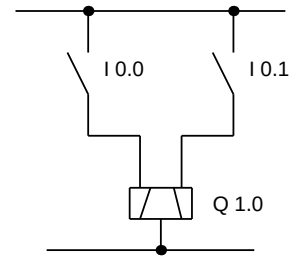
8.1.2 Set/Reset Operations

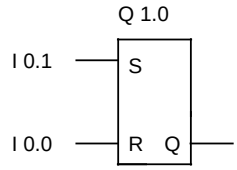
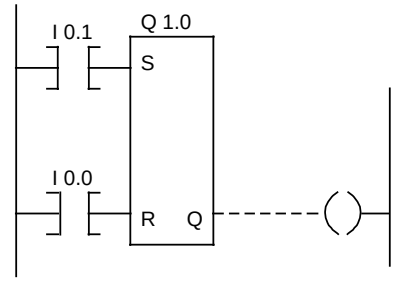
Set/reset operations store the result of logic operation (RLO) formed in the processor. The stored RLO represents the signal state of the addressed operand. Storage can be dynamic (assignment) or static (set and reset). Table 8-2 provides an overview of the set/reset operations. Examples follow the table.

Table 8-2. Overview of the Set/Reset Operations

Operation	Operand		Meaning
S			**Set** The first time the program is scanned with RLO =" 1", signal state "1" is assigned to the addressed operand. An RLO change does not affect this status.
R			**Reset** The first time the program is scanned with RLO = "1", signal state 0 is assigned to the addressed operand. An RLO change does not affect this status.
=			**Assign** Every time the program is scanned, the current RLO is assigned to the addressed operand.
	ID	**Parameter**	
	I	0.0 to 127.7	
	Q	0.0 to 127.7	
	F	0.0 to 255.7	

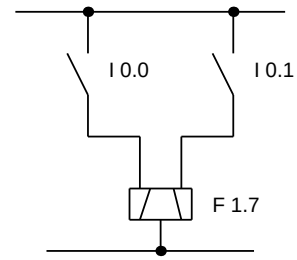
Flip-Flop for a Latching Signal Output (reset dominant)

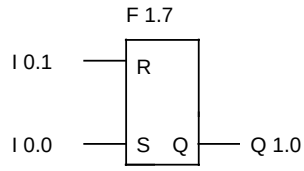
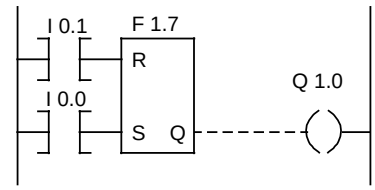
Example	Circuit Diagram
A" 1" at input I 0.1 sets flip-flop Q 1.0 (signal state "1"). If the signal state at input I 0.1 changes to" 0", the state of output Q 1.0 is maintained, i.e., the signal is latched. A "1" at input I 0.0 resets the flip-flop (signal state 0"). When the "SET" signal (input I 0.1) and the "RESET" signal (input I 0.0) are applied at the same time, the scanning operation that was programmed last (in this case A I 0.0) is in effect during processing of the rest of the program. In this example, resetting output Q 1.0 has priority.	

STL	CSF	LAD
A I 0.1 S Q 1.0 A I 0.0 R Q 1.0 **NOP 0 ***		

* **NOP 0** "NOP 0" is necessary if the program is to be represented in LAD or CSF form on programmers with a screen. During programming in LAD and CSF, such "NOP 0" operations are allotted automatically.

RS Flip-Flop with Flags (set dominant)

Example	Circuit Diagram
A "1" at input I 0.0 sets flip-flop F 1.7 (signal state "1"). If the signal state at input I 0.0 changes to "0", the state of flag F 1.7 is maintained, i.e., the signal is latched. A "1" at input 0.1 resets the flip-flop (signal state "0"). If the signal state at input I 0.1 changes to "0", flag F 1.7 retains signal state "0". If both inputs have a "1" signal state, the flip-flop is set (set dominant). The signal state of the flag is scanned and transferred to output Q 1.0.	

STL	CSF	LAD
A I 0.1 R F 1.7 A I 0.0 S F 1.7 A F 1.7 = Q 1.0		

8.1.3 Load and Transfer Operations

Use load and transfer operations to do the following tasks.

- Exchange information between various operand areas
- Prepare time and count values for further processing
- Load constants for program processing

Information flows indirectly via accumulators (ACCU 1 and ACCU 2). The accumulators are special registers in the programmable controller that serve as temporary storage. They are each 16 bits long. The accumulators are structured as shown in Figure 8-1.

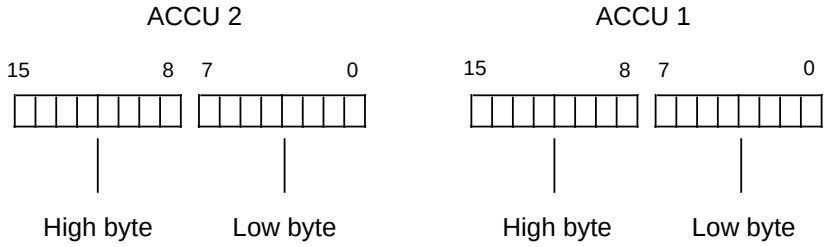


Figure 8-1. Accumulator Structure

You can load and transfer permissible operands in bytes or words. For exchange in bytes, information is stored right-justified, i.e., in the low byte.
The remaining bits are set to zero.
You can use various operations to process the information in the two accumulators.

Load and transfer operations are executed independently of condition codes. Execution of these operations does not affect the condition codes.
You can program load and transfer operations graphically only in combination with timer or counter operations; otherwise you can represent them only in STL form.

Table 8-3 provides an overview of the load and transfer operations. Examples follow the table.

Table 8-3. Overview of Load and Transfer Operations

Operation	Operand		Meaning
L			**Load** The operand contents are copied into ACCU 1 regardless of the RLO. The RLO is not affected.
T			**Transfer** The contents of ACCU 1 are assigned to an operand regardless of the RLO. The RLO is not affected.
	ID		**Parameter**
	IB		0 to 127
	IW		0 to 126
	QB		0 to 127
	QW		0 to 126
	FY		0 to 255
	FW		0 to 254
	DR		0 to 255
	DL		0 to 255
	DW		0 to 255
	T[1]		0 to 127
	C[1]		0 to 127
	PB/PY*		0 to 127
	PW		0 to 126
	KM[1]		random bit pattern (16 bits)
	KH[1]		0000 to FFFF
	KF[1]		- 32768to +32767
	KY[1]		0 to 255 per byte
	KB[1]		0 to 255
	KS[1]		any 2 alphanumeric characters
	KT[1]		0.0 to 999.3
	KC[1]		0 to 999
LD			**Load in BCD** Binary times and counts are loaded into ACCU 1 in BCD code regardless of the RLO.
	ID		**Parameter**
	T		0 to 127
	Z		0 to 127

1 These operands cannot be used for transfer.

Load Operation

During loading, information is copied from a memory area, e.g., from the PII, into ACCU 1.
The previous contents of ACCU 1 are shifted to ACCU 2.
The original contents of ACCU 2 are lost.

Example: Two consecutive bytes (IB7 and IB8) are loaded from the PII into the accumulator. Loading does not change the PII (see Figure 8-2).

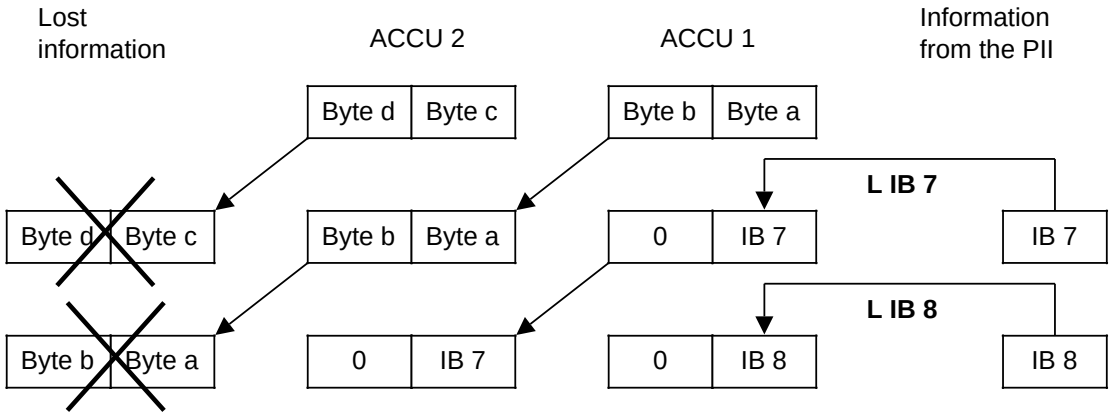


Figure 8-2. Execution of the Load Operation

Transfer Operation

During transfer, information from ACCU 1 is copied into the addressed memory area, e.g., into the PIQ.
This transfer does not affect the contents of ACCU 1.

Example: Figure 8-3 shows how byte a, the low byte in ACCU 1, is transferred to QB5.

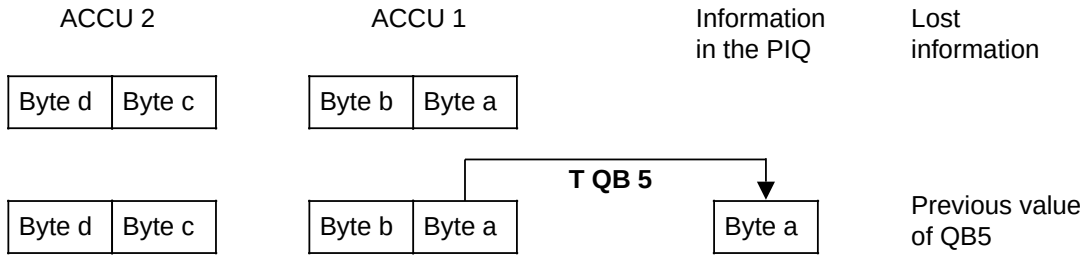
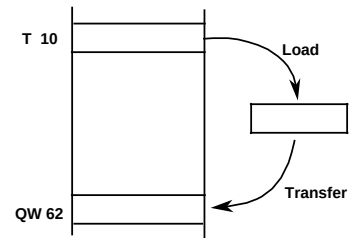


Figure 8-3. Transferring a Byte

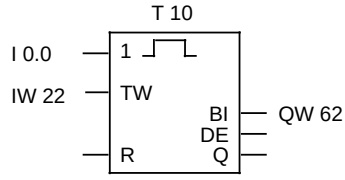
Loading and Transferring a Time (See also Timer and Counter Operations)

Example	Representation
During graphic input, QW62 is assigned to output BI of a timer. The programmer automatically stores the corresponding load and transfer operation in the control program. Thus the contents of the memory location addressed with T 10 are loaded into ACCU 1. Afterwards, the contents of the accumulator are transferred to the process image addressed with QW62. In this example, you can see timer T 10 at QW62 in binary code. Outputs BI and DE are digital outputs. The time at output BI is in binary code. The time at output DE is in BCD code with time base.	

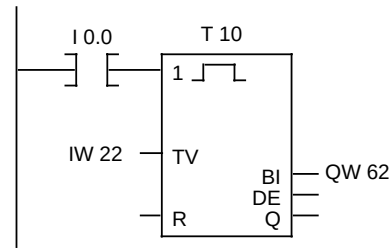
STL

```
A     I    0.0
L     IW   22
SP    T    10
NOP 0
L     T    10
T     QW   62
NOP 0
NOP 0
```

CSF



LAD

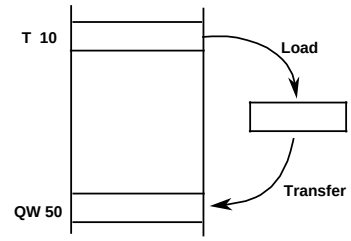


Note

The timers are divided into two areas T 0 to T 63 and T 64 to T 127. Please note that the timers are updated at different times:

- Timers T 0 to T 63 are updated by the operating system always when you scan the timer in the control program.
- Timers T 64 to T 127 are updated by the operating system always prior to processing of OB1. The timers are not updated during the start-up organization blocks OB21/OB22.

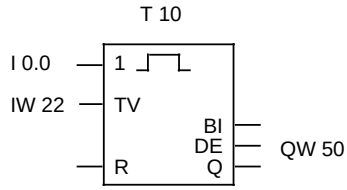
Loading and Transferring a Time (Coded)

Example	Representation
The contents of the memory location addressed with T 10 are loaded into the accumulator in BCD code. Then a transfer operation transfers the accumulator contents to the process image memory location addressed by QW50. A coding operation is possible only indirectly for the graphic representation forms LAD and CSF by assigning an address to output DE of a timer or counter location. However, this operation can be entered with a separate statement with STL.	

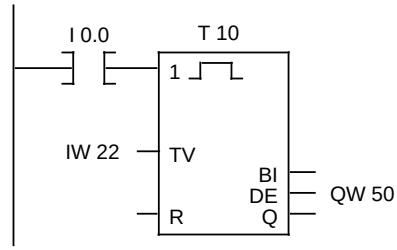
STL

```
A     I    0.0
L     IW   22
SP    T    10
NOP 0
NOP 0
LD    T    10
T     QW   50
NOP 0
```

CSF



LAD



Note

The timers are divided into two areas T 0 to T 63 and T 64 to T 127. Please note that the timers are updated at different times:

- Timers T 0 to T 63 are updated by the operating system always when you scan the timer in the control program.
- Timers T 64 to T 127 are updated by the operating system always prior to processing of OB1. The timers are not updated during the start-up organization blocks OB21/OB22.

8.1.4 Timer Operations

The program uses timer operations to implement and monitor chronological sequences. Table 8-4 provides an overview of timer operations. Examples follow the table.

Table 8-4. Overview of Timer Operations

Operation	Operand		Meaning
SP			**Pulse Timer** The timer is started on the leading edge of the RLO. When the RLO is "0", the timer is set to "0". Scans result in signal state "1" as long as the timer is running.
SE			**Extended Pulse Timer** The timer is started on the leading edge of the RLO. When the RLO is "0", the timer is not affected. Scans result in signal state "1" as long as the timer is running.
SD			**On-Delay Timer** The timer is started on the leading edge of the RLO. When the RLO is "0", the timer is set to "0". Scans result in signal state "1" when the timer has run out and the RLO is still pending at the input.
SS			**Stored On-Delay Timer** The timer is started on the leading edge of the RLO. When the RLO is "0", the timer is not affected. Scans result in signal state "1" when the timer has run out. The signal state becomes "0" when the timer is reset with the "R" operation.
SF			**Off-Delay Timer** The timer is started on the trailing edge of the RLO. When the RLO is "1", the timer is set to its initial value. Scans result in signal state "1" as long as the RLO at the input is "1" or the timer is still running.
R			**Reset Timer** The timer is reset to its initial value as long as the RLO is "1". When the RLO is "0", the timer is not affected. Scans result in signal state "0" as long as the timer is reset or has not been started yet.
ID T			**Parameter** 0 to 127

Updating the Timers

The operating system of the S5-95F ensures that the same times or timer statuses are present in both subunits when a timer is scanned.

Note

The timers are divided into two areas T 0 to T 63 and T 64 to T 127. Please note that the timers are updated at different times:

- Timers T 0 to T 63 are updated by the operating system always when you scan the timer in the control program.
- Timers T 64 to T 127 are updated by the operating system always prior to processing of OB1. The timers are not updated during the start-up organization blocks OB21/OB22.

Loading a Time

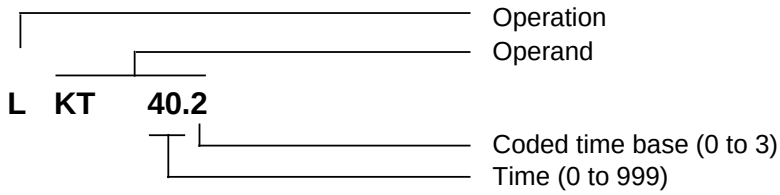
Timer operations call internal timers.
When a timer operation is started, the word in ACCU 1 is used as a time value. You must therefore first specify time values in the accumulator.

You can load a timer with any of the following data types:

KT	constant time value	
	or	
DW	data word	These data types must be in BCD code.
IW	input word	
QW	output word	
FW	flag word	

Loading a Constant Time Value:

The following example shows how you can load a time value of 40 s.



Key for Time Base

Base	0	1	2	3
Factor	0.01 s	0.1 s	1 s	10 s

Example: KT 40.2 corresponds to 40 x 1 s.

Tolerance:

The time tolerance is equivalent to the time base.

Examples	Operand	Time Interval		
Possible settings for the time 40 s	KT 400.1	400 × 0.1 s -	0.1 s	39.9 s ... 40 s
	KT 40.2	40 × 1 s -	1 s	39 s ... 40 s
	KT 4.3	4 × 10 s -	10 s	30 s ... 40 s

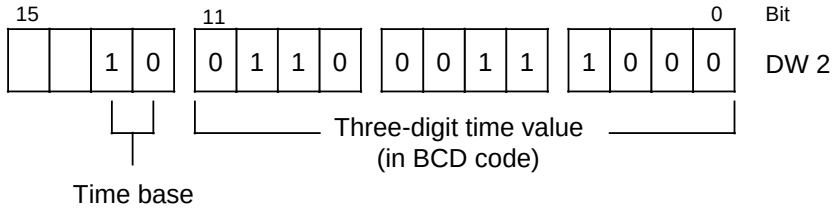
Note

Always use the smallest time base possible.

Loading a Time as an Input, Output, Flag, or Data Word

Load Statement: **L DW 2**

The time 638 s is stored in data word DW2 in BCD code.
Bits 14 and 15 are insignificant for the time value.



Key for Time Base:

Base	0 0	0 1	1 0	1 1
Factor	0.01 s	0.1 s	1 s	10 s

You can also use the control program to write to data word DW2.

Example: Store the value 270 x 100 ms in data word DW2 of data block DB3.

```
C   DB 3
L   KT 270.1
T   DW2
```

Output of the Current Time

You can use a load operation to put the current time into ACCU 1 and process it further from there (see Figure 8-4).

Use the "Load in BCD" operation for digital display output.

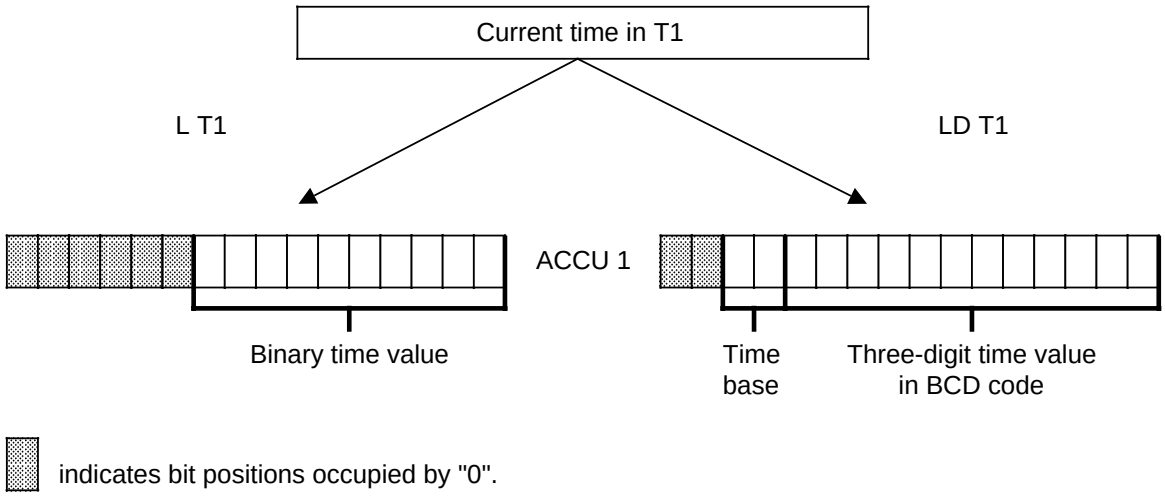
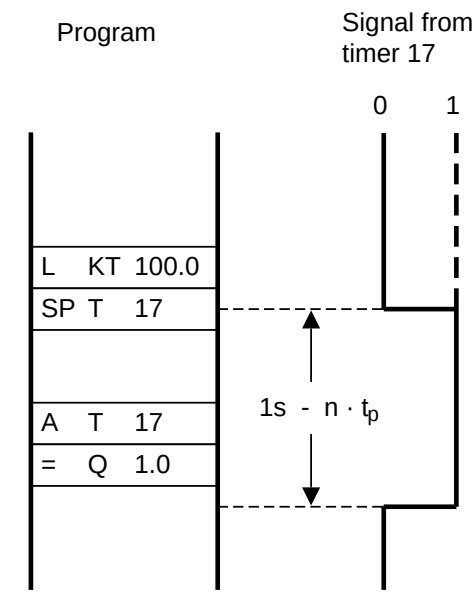


Figure 8-4. Output of the Current Time (Example)

Starting a Timer

Example:

Schematic Representation	Explanation
 n: number of program scanning cycles t_p: program scan time	The schematic shows the "n^{th} + 1" processing cycle since timer T 17* was started. Although the timer ran out shortly after the statement "= Q 1.0", output Q 1.0 remains set. The change is not considered until the next program scanning cycle. * KT 100.0 is equal to 1 s.

Except for "Reset timer", all timer operations are started only when there is an edge change. The RLO changes from "0" to "1".

If there is an edge change while the timer is running, the timer is reset to its initial value and restarted.

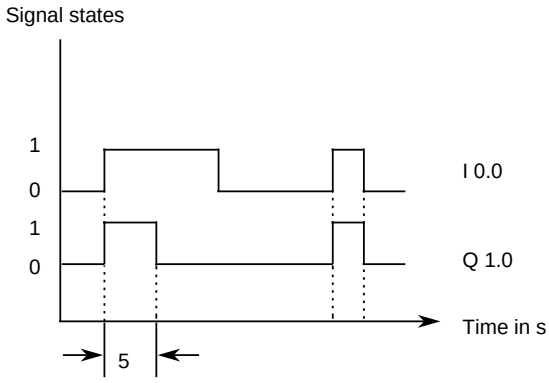
The signal state of a timer can be scanned with Boolean logic operations.

Pulse

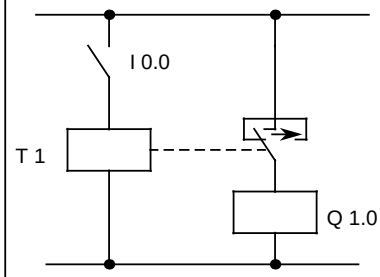
Example: Output Q 1.0 is set when the signal state at input I 0.0 changes from “0” to “1”.

However, the output should not remain set longer than 5 s.

Timing Diagram



Circuit Diagram

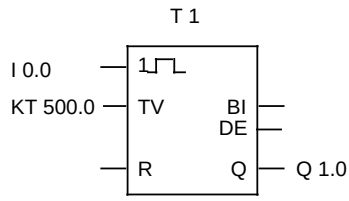


T 1: Time relay with transitional NO contact

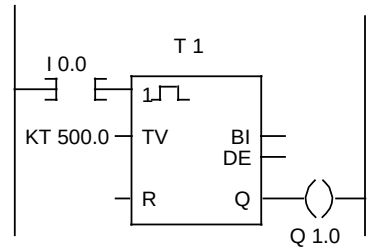
STL

```
A     I    0.0
L     KT  500.0
SP    T    1
NOP 0
NOP 0
NOP 0
A     T    1
=     Q    1.0
```

CSF



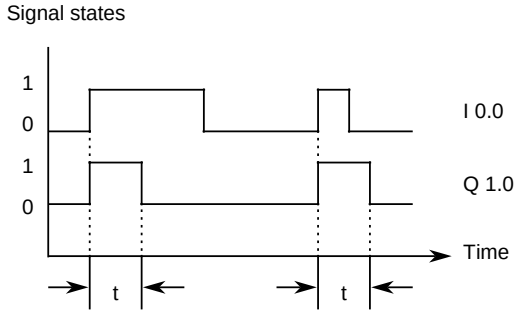
LAD



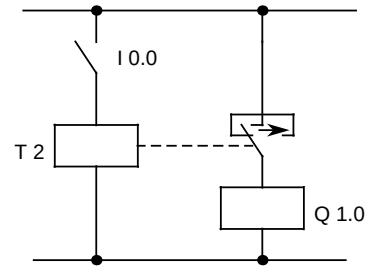
Extended pulse

Example: Output Q 1.0 is set for a specific time when the signal at input I 0.0 changes to “1”. The time is indicated in IW16.

Timing Diagram



Circuit Diagram

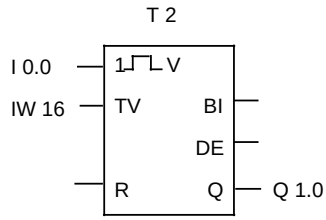


T 2: Time relay with pulse shaper

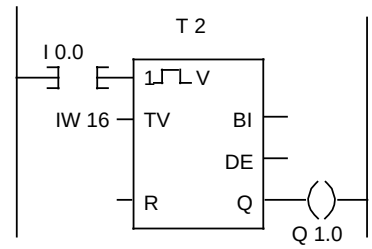
STL

```
A     I    0.0
L     IW   16
SE    T    2
NOP 0
NOP 0
NOP 0
A     T    2
=     Q    1.0
```

CSF



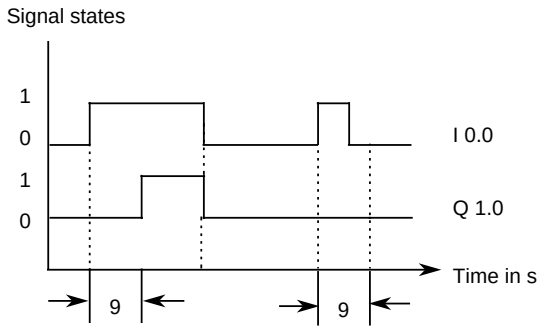
LAD



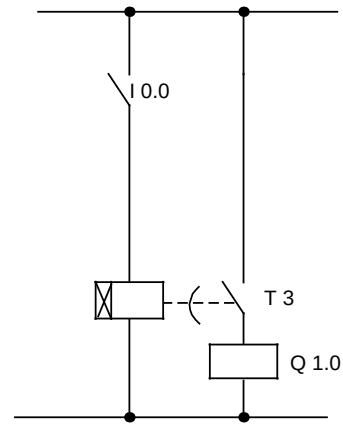
On-Delay

Example: Output Q 1.0 is set 9 s after input I 0.0 and remains set as long as the input carries signal "1".

Timing Diagram



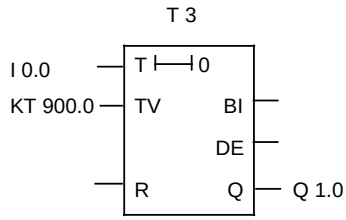
Circuit Diagram



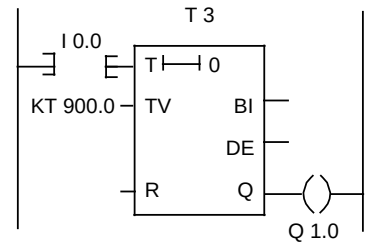
STL

```
A     I     0.0
L     KT  900.0
SD    T   3
NOP 0
NOP 0
NOP 0
A     T   3
=     Q     1.0
```

CSF



LAD

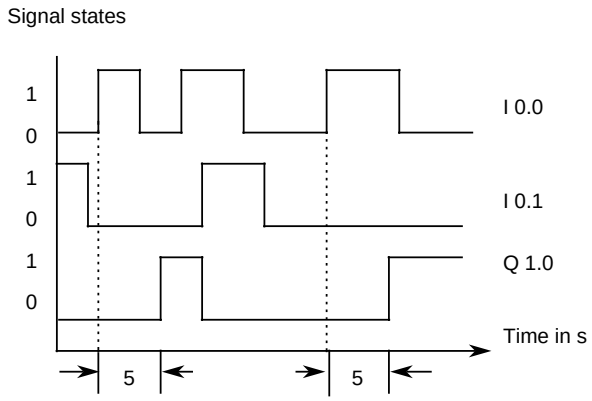


Stored On-Delay and Reset

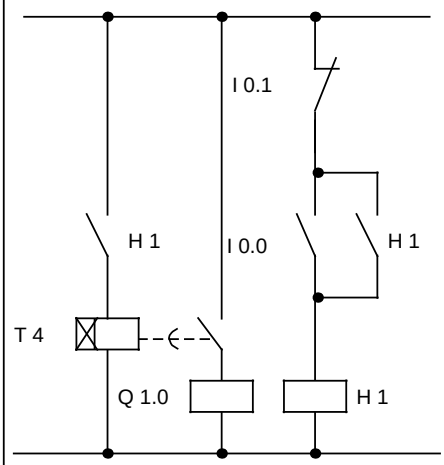
Example: Output Q 1.0 is set 5 s after I 0.0.

Further changes in the signal state at input I 0.0 do not affect the output.
Input I 0.1 resets timer T 4 to its initial value and sets output Q 1.0 to zero.

Timing Diagram



Circuit Diagram

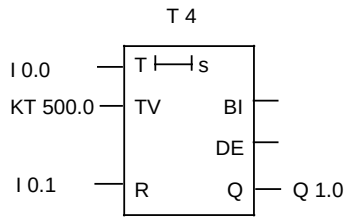


H 1: Auxiliary relay

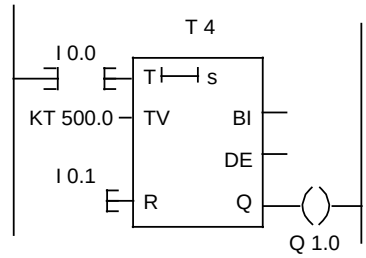
STL

```
A     I    0.0
L     KT   500.0
SS    T    4
A     I    0.1
R     T    4
NOP 0
NOP 0
A     T    4
=     Q    1.0
```

CSF



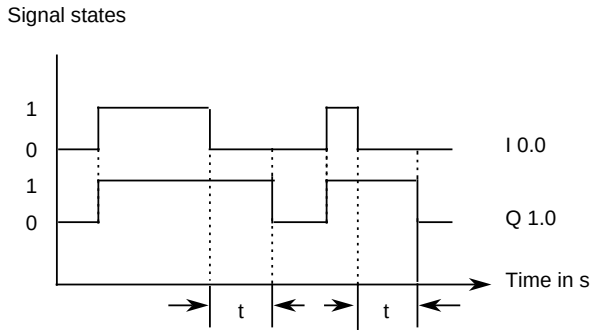
LAD



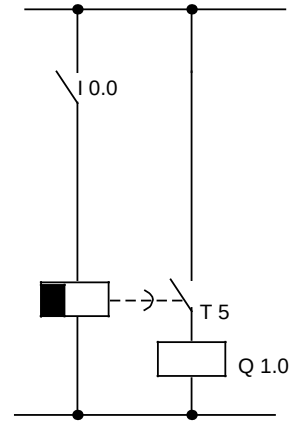
Off-Delay

Example When input I 0.0 is reset, output Q 1.0 is set to zero after a certain delay (t). The value in FW14 specifies the delay time.

Timing Diagram



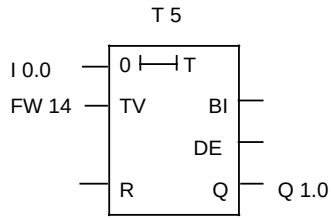
Circuit Diagram



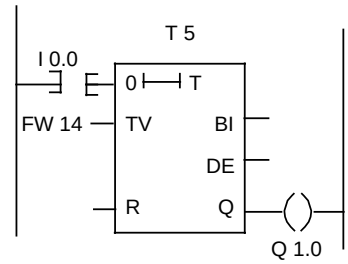
STL

```
A     I    0.0
L     FW   14
SF    T    5
NOP 0
NOP 0
NOP 0
A     T    5
=     Q    1.0
```

CSF



LAD



8.1.5 Counter Operations

The programmable controller uses counter operations to handle counting jobs. Counters can count up and down. The counting range is from 0 to 999 (three decades). Table 8-5 provides an overview of the counter operations. Examples follow the table.

Table 8-5. Overview of Counter Operations

Operation	Operand		Meaning
S			**Set Counter** The counter is set on the leading edge of the RLO.
R			**Reset Counter** The counter is set to zero as long as the RLO is "1".
CU			**Count Up** The count is incremented by 1 on the leading edge of the RLO. When the RLO is "0", the count is not affected.
CD			**Count Down** The count is decremented by 1 on the leading edge of the RLO. When the RLO is "0", the count is not affected.
	ID Z		**Parameter** 0 to 127

Loading a Count

Counter operations call internal counters.

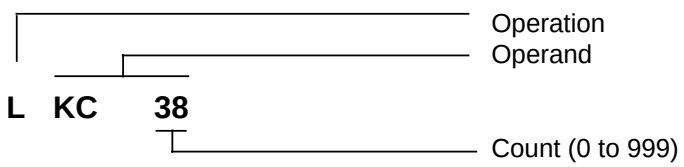
When a counter is set, the word in ACCU 1 is used as a count. You must therefore first store counts in the accumulator.

You can load a count with any of the following data types:

KC	constant count	
	or	
DW	data word	The data for these words must be in BCD code.
IW	input word	
QW	output word	
FW	flag word	

Loading a Constant Count

The following example shows how the count 38 is loaded.

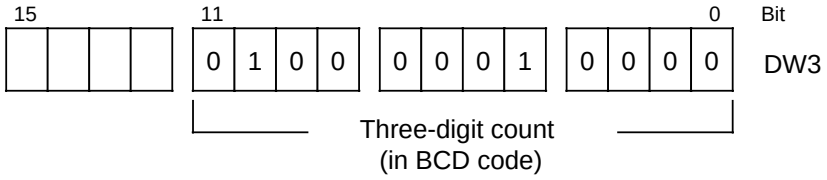


Loading a Count as an Input, Output, Flag, or Data Word

Load statement: **L DW 3**

The count 410 is stored in data word DW3 in BCD code.

Bits 12 to 15 are insignificant for the count.

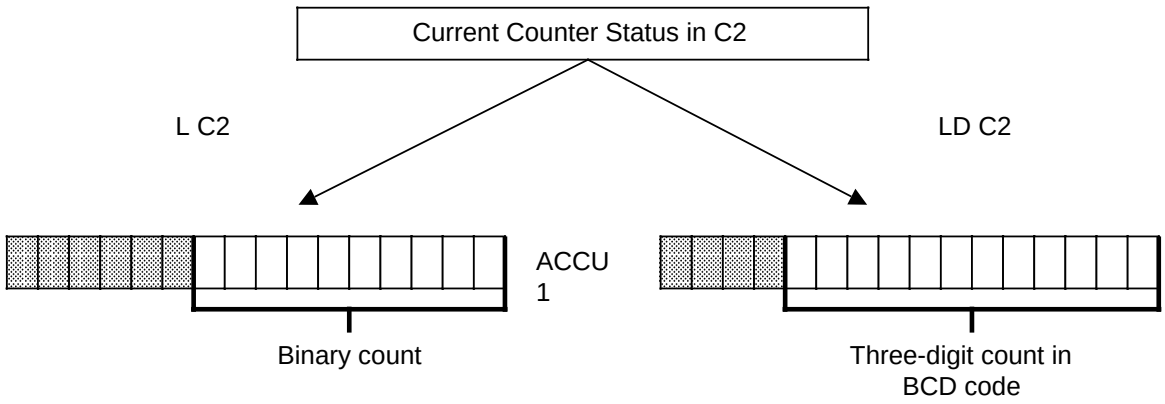


Scanning the Counter

Use Boolean logic operations to scan the counter status (e.g., A Cx). As long as the count is not zero, the scan result is signal state “1”.

Output of the Current Counter Status

You can use a load operation to put the current counter status into ACCU 1 and process it further from there. The “Load in BCD” operation outputs a digital display (see Figure 8-5).



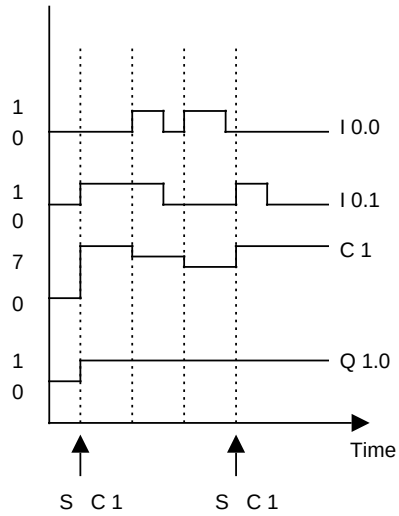
Indicates bit positions occupied by "0".

Figure 8-5. Output of the Current Counter Status (Example)

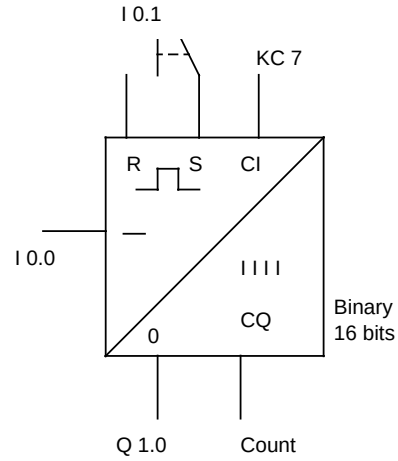
Setting a Counter “S” and Counting Down “CD”

Example: When input I 0.1 is switched on (set), counter 1 is set to count 7. Output Q 1.0 is now “1”. Every time input I 0.0 is switched on (count down), the count is decremented by 1. The output is set to “0” when the count is “0”.

Timing Diagram



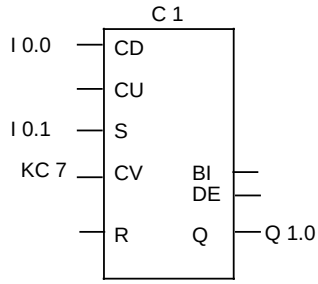
Circuit Diagram



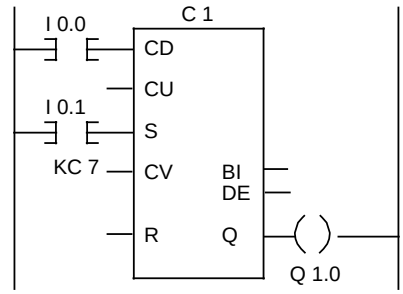
STL

```
A     I    0.0
CD    C    1
NOP   0
A     I    0.1
L     KC   7
S     C    1
NOP   0
NOP   0
NOP   0
A     C    1
=     Q    1.0
```

CSF



LAD

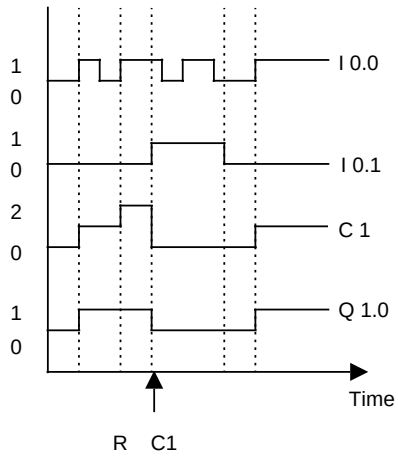


Resetting a Counter “R” and Counting Up “CU”

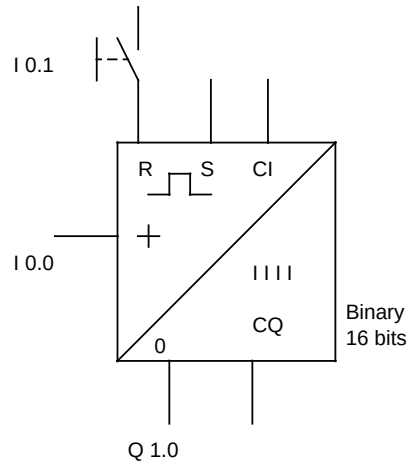
Example: When input I 0.0 is switched on, the count in counter 1 is incremented by 1. As long as a second input (I 0.1) is "1", the count is reset to "0".

The A C 1 operation results in signal state "1" at output Q 1.0 as long as the count is not "0".

Timing Diagram



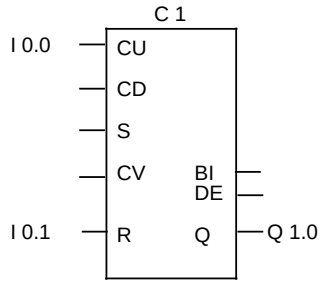
Circuit Diagram



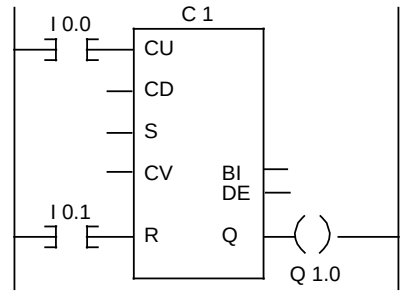
STL

```
A     I    0.0
CU    C    1
NOP   0
NOP   0
NOP   0
A     I    0.1
R     C    1
NOP   0
NOP   0
A     C    1
=     Q    1.0
```

CSF



LAD



8.1.6 Comparison Operations

Comparison operations compare the contents of the two accumulators. The comparison does not change the accumulators' contents. Table 8-6 provides an overview of the comparison operations. An example follows the table.

Table 8-6. Overview of Comparison Operations

Operation	Operand		Meaning
! = F			**Compare for "equal to"** The contents of the two accumulators are interpreted as bit patterns and scanned to see if they are equal.
> < F			**Compare for "not equal to"** The contents of the two accumulators are interpreted as bit patterns and compared to see if they are not equal.
> F			**Compare for "greater than"** The contents of the two accumulators are interpreted as fixed-point numbers. They are compared to see if the operand in ACCU 2 is greater than the operand in ACCU 1.
> = F			**Compare for "greater than or equal to"** The contents of the two accumulators are interpreted as fixed-point numbers. They are compared to see if the operand in ACCU 2 is greater than or equal to the operand in ACCU 1.
< F			**Compare for "less than"** The contents of the two accumulators are interpreted as fixed-point numbers. They are compared to see if the operand in ACCU 2 is less than the operand in ACCU 1.
< = F			**Compare for "less than or equal to"** The contents of the two accumulators are interpreted as fixed-point numbers. They are compared to see if the operand in ACCU 2 is less than or equal to the operand in ACCU 1.

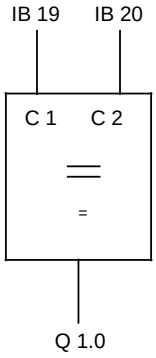
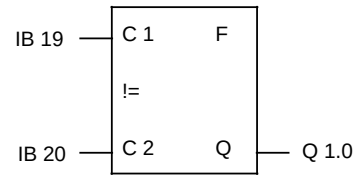
Processing Comparison Operations

To compare two operands, load them consecutively into the two accumulators. Execution of the operations is independent of the RLO.
The result is binary and is available as RLO for further program processing. If the comparison is satisfied, the RLO is “1”. Otherwise it is “0”.

Executing the comparison operations sets the condition codes (see section 8.4).

> **Note**
>
> When using comparison operations, make sure the operands have the same number format.

Example: The values of input bytes IB19 and IB20 are compared. If they are equal, output Q 1.0 is set.

Circuit Diagram	STL	CSF/LAD
	L IB 19 L IB 20 !=F = Q 1.0	

8.1.7 Arithmetic Operations

Arithmetic operations interpret the contents of the accumulators as fixed-point numbers and manipulate them. The result is stored in ACCU 1. Table 8-7 provides an overview of the arithmetic operations. An example follows the table.

Table 8-7. Overview of Arithmetic Operations

Operation	Operand		Meaning
+ F			**Addition** The contents of both accumulators are added.
- F			**Subtraction** The contents of ACCU 1 are subtracted from the contents of ACCU 2.

Integral function blocks are available for multiplication and division (see section 9.2).

Processing an Arithmetic Operation

Before an arithmetic operation is executed, both operands must be loaded into the accumulators.

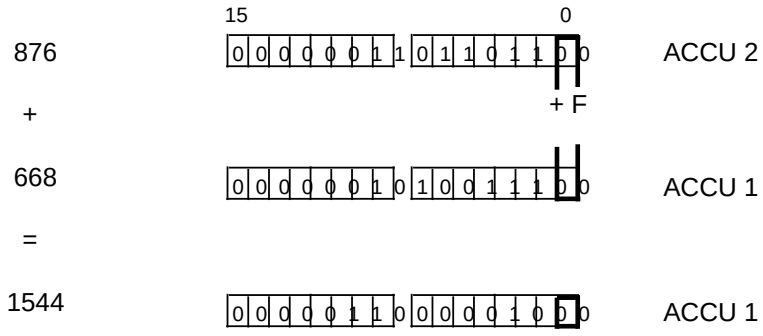
> **Note**
>
> When using arithmetic operations, make sure the operands have the same number format.

Arithmetic operations are executed independently of the RLO. The result is available in ACCU 1 for further processing. The contents of ACCU 2 are not changed.

These operations do not affect the RLO. The condition codes are set according to the results.

STL	Explanation
`L C 3`	The value of counter 3 is loaded into ACCU 1.
`L C 1`	The value of counter 1 is loaded into ACCU 1. The previous contents of ACCU 1 are shifted to ACCU 2.
`+ F`	The contents of the two accumulators are interpreted as 16-bit fixed-point numbers and added.
`T QW 12`	The result, contents of ACCU 1, is transferred to output word QW12.

Numeric Example



8.1.8 Block Call Operations

Block call operations specify the sequence of a structured program. Table 8-8 provides an overview of the block call operations. Examples follow the table.

Table 8-8. Overview of Block Call Operations

Operation	Operand	Meaning
JU		**Jump unconditionally** Program scanning continues in a different block regardless of the RLO. The RLO is not affected.
JC		**Jump conditionally** Program scanning jumps to a different block when the RLO is "1". Otherwise program scanning continues in the same block. The RLO is set to "1".
ID	**Parameter**	
OB	0, 1, 3 ... 255*	
PB	0 to 255	
FB	0 to 255	
SB	0 to 255	
C		**Call a data block** A data block is activated regardless of the RLO. Program scanning is not interrupted. The RLO is not affected.
G		**Generate and delete a data block**** An area is set up in the RAM to store data regardless of the RLO.
ID	**Parameter**	
DB	2 to 255 for C DB and 2 to 251, 255*** for G DB	
BE		**Block end** The current block is terminated regardless of the RLO. Program scanning continues in the block in which the call originated. The RLO is "carried along" but not affected. BE is always the last statement in a block.
BEU		**Block end, unconditional** The current block is terminated regardless of the RLO. Program scanning continues in the block in which the call originated. The RLO is "carried along" but not affected.
BEC		**Block end, conditional** When the RLO is "1", the current block is terminated. Program scanning continues in the block in which the call originated. During the block change, the RLO remains "1". If the RLO is "0", the operation is not executed. The RLO is set to "1" and linear program scanning continues.

* If using the PG 615, select "SYS. OPS. Y" in the presets menu. Also note that certain OBs are assigned by the operating system.

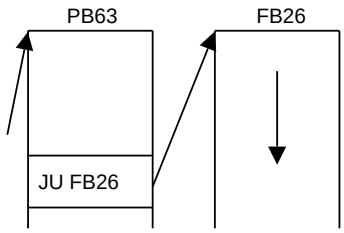
** The length of the DB must be loaded into ACCU 1 before execution of the operation. A length of 0 makes the DB invalid.

*** Data blocks DB0 and DB1 are reserved for special functions.

Unconditional Block Call “JU”

One block is called within another block, regardless of conditions.

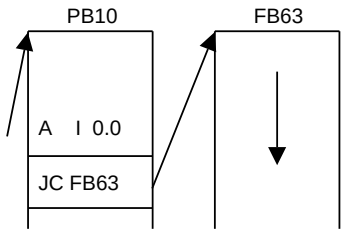
Example: A special function has been programmed in FB26. It is called at several locations in the program, e.g., in PB63, and processed.

Program Sequence	STL	Explanation
	. . . . . JU FB 26 .	The “JU FB26” statement in program block PB63 calls function block FB26.

Conditional Block Call “JC”

One block is called within another block when the previous condition has been satisfied (RLO = “1”).

Example: A special function has been programmed in FB63. It is called and processed under certain conditions, e.g., in PB10.

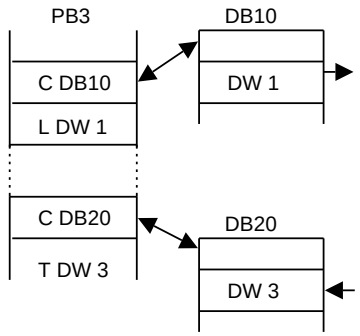
Program Sequence	STL	Explanation
	. . . S F 1.0 A I 0.0 JC FB 63 .	The "JC FB63" statement in program block PB10 calls function block FB63 if input I 0.0 is "1".

Call a Data Block “C DB”

Data blocks are always called unconditionally. All data processed following the call refers to the data block that has been called.

This operation cannot generate new data blocks. Blocks that are called must be programmed or created before program scanning.

Example: Program block PB3 needs information that has been programmed as data word DW1 in data block DB10. Other data, e.g., the result of an arithmetic operation, is stored as data word DW3 in data block DB20.

Program Sequence	STL	Explanation
	C DB 10 L DW 1 . . . . C DB 20 T DW 3	The information from data word DW1 in data block DB10 is loaded into the accumulator. The contents of ACCU 1 are stored in data word DW3 of data block DB20.

Generating and Deleting a Data Block

The “G DB x” statement does not call a data block. Instead, it generates a new block. If you want to use the data in this data block, call it with the “C DB” statement.

Before the “G DB” statement, indicate in ACCU 1 the number of data words the block is to have (see the example below).

If you specify zero as the data block length, the data block in question is deleted, i.e., it is removed from the address list. It is considered nonexistent.

Note

The block is stored in memory and is designated as invalid until the programmable controller memory is compressed (see section 7.5.3).

If you try to set up a data block that already exists, the “G DB x” statement is not executed.

A data block can be a maximum of 256 data words (DW0 to 255) in length.

Generating a Data Block

Example	STL	Explanation
Generate a data block with 128 data words without the aid of a programmer.	L KF + 128 **G DB 5**	The constant fixed-point number +127 is loaded into ACCU 1. At the same time, the old contents of ACCU 1 are shifted to ACCU 2. Data block 5 is generated with a length of 128 data words (0000) in the RAM of the PLC and entered in the block address list. The next time the “G DB5” operation is processed, it has no effect if the contents of ACCU 1 are not 0.

Deleting a Data Block

Example	STL	Explanation
Delete a data block that is no longer needed.	L KF + 0 G DB 5	The constant fixed-point number +0 is loaded into ACCU 1. At the same time, the old contents of ACCU 1 are shifted to ACCU 2. Data block 5, which must be in the RAM of the PLC, is declared invalid and removed from the block address list.

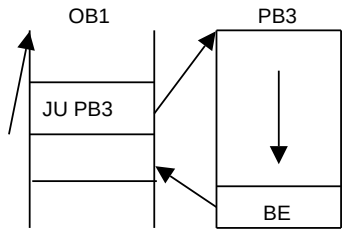
Block End “BE”

The “BE” operation terminates a block. Data blocks do not need to be terminated. “BE” is always the last statement in a block.

In structured programming, program scanning jumps back to the block where the call for the current block was made.

Boolean logic operations cannot be continued in a higher-order block.

Example: Program block PB3 is terminated by the “BE” statement.

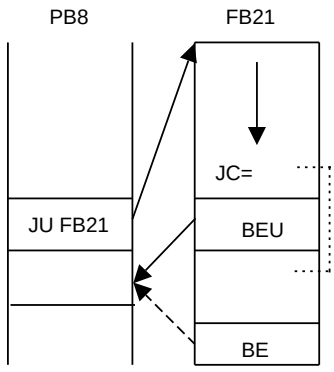
Program Sequence	STL	Explanation
	. . . . . . BE	The "BE" statement terminates program block PB3 and causes program scanning to return to organization block OB1.

Unconditional Block End "BEU"

The "BEU" operation causes a return within a block. However, jump operations can bypass the "BEU" operation in function blocks (see sections 8.2.10 and 8.3.4).

Binary logic operations cannot be continued in a higher-order block.

Example: Scanning of function block FB21 is terminated regardless of the RLO.

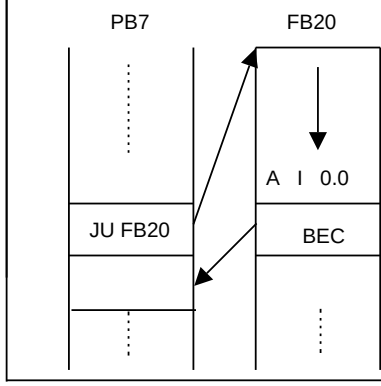
Program Sequence	STL	Explanation
	. . . . JC=M001 BEU M001: . . . . BE	The "BEU" statement causes program scanning to leave function block FB21 and return to program block PB8.

Conditional Block End "BEC"

The "BEC" operation causes a return within a block if the previous condition has been satisfied (RLO = 1).

Otherwise, linear program scanning is continued with RLO "1".

Example: Scanning of program block FB20 is terminated if the RLO = "1".

Program Sequence	STL	Explanation
	. . . . A I 0.0 BEC	The "BEC" statement causes program scanning to return to program block PB7 from function block FB20 if input I 0.0 is "1".

8.1.9 Other Operations

Table 8-9 lists other basic operations. Explanations follow the table.

Table 8-9. Overview of Other Operations

Operation	Operand	Meaning
STP		**Stop at the end of program scanning (in OB1)** Current program scanning is terminated. The PIQ is read out. Then the PLC goes into the STOP mode.
NOP 0		**"No" Operation** Sixteen bits in the RAM are set to "0".
NOP 1		**"No" Operation** Sixteen bits in the RAM are set to "1".
BLD		"BLD" means a display generation operation for the programmer.
ID	**Parameter** 130, 131, 132, 133, 255	

Note

These operations can be programmed in STL form only.

STOP Operation

The “STP” operation puts the programmable controller into the STOP mode. This can be desirable for time-critical system circumstances or when a programmable controller error occurs.

After the statement is processed, the control program is scanned to the end, regardless of the RLO. Afterwards the programmable controller goes into the STOP mode with the error ID “STS”. You can restart the programmable controller with the mode selector (STOP to RUN) or with a programmer.

> **Safety Note**
>
> A STOP initiated by the ***”STP” operation*** can very easily be cancelled by means of a programmer input (even inadvertently). A STOP produced by the ”STP” operation is therefore ***not a failsafe STOP condition*** (see section 18.16).

“NOP” (No Operations)

The “NOP” operations reserve or overwrite memory locations.

Display Generation Operations

“BLD” display generation operations divide program parts into segments within a block.

“NOP” operations and display generation operations are significant only for the programmer when representing the STEP 5 program.

The programmable controller does not execute any operation when these statements are processed.

8.2 Supplementary Operations

Supplementary operations extend the operations set. However, compared to basic operations, which can be programmed in all blocks, supplementary operations have the following limitations.

- They can be programmed in function blocks only.
- In most cases, they can be represented in STL form only. With STEP 5 as of version V 6.0, however, you can also program some operations in CSF (see programmer manual).

The following sections describe the supplementary operations.

8.2.1 Load Operation

As with the basic load operations, the supplementary load operation copies information into the accumulator. Table 8-10 explains the load operation. An example follows the table.

Table 8-10. Load Operation

Operation	Operand		Meaning
L			**Load** A word from the system data is loaded into ACCU 1 regardless of the RLO.
	ID RS		**Parameter** 0 to 255

Example	STL	Explanation
Read out the OB 13 time interval from the system data.	. . . L RS 97 . . .	Load ACCU 1 with OB 13 time interval (multiply value by time basis of 10 ms).

8.2.2 Enable Operation

You can use the enable operation (FR) to execute the following operations even without an edge change.

- Start a timer
- Set a counter
- Count up and down

Table 8-11 presents the enable operation. An example follows the table.

Table 8-11. Enable Operation

Operation	Operand		Meaning
FR			**Enable a Timer/Counter** Timers and counters are enabled on the leading edge of the RLO. This operation restarts a timer, sets a counter, or causes a counter to count up or down when the RLO "1" is pending at the "Start" operation.
	ID	**Parameter**	
	T	0 to 127	
	C	0 to 127	

Example	STL	Explanation
Input I 0.0 starts a timer T 2 as an extended pulse (pulse width 50 s). This timer sets output Q 1.0 for the duration of the pulse.	A I 0.0 L KT 500.1 SE T 2 A T 2 = Q 1.0	Start a timer T 2 as an extended pulse. Output Q 1.0 is set for 50s.
If output Q 1.1 is reset repeatedly, the timer should also be restarted repeatedly.	A Q 1.1 **FR T 2** BE	If output Q 1.1 is set (positive edge change of the RLO) during the time in which input I 0.0 is set, timer T2 is restarted. Output Q 1.0 therefore remains set at the restarted time or is reset. If input I 0.0 is not set during the edge change of output Q 1.1, the timer is not restarted.

8.2.3 Bit Test Operations

Bit test operations scan digital operands bit by bit and affect them. Bit test operations must always be at the beginning of a logic operation. Table 8-12 provides an overview of these operations.

Table 8-12. Overview of Bit Operations

Operation	Operand		Meaning
TB			**Test a bit for signal state "1"** A single bit is scanned regardless of the RLO. The RLO is affected according to the bit's signal state (see Table 8-13).
TBN			**Test a bit for signal state "0"** A single bit is scanned regardless of the RLO. The RLO is affected according to the bit's signal state (see Table 8-13).
SU			**Set a bit unconditionally** The addressed bit is set to "1" regardless of the RLO. The RLO is not affected.
RU			**Reset a bit unconditionally** The addressed bit is set to "0" regardless of the RLO. The RLO is not affected.
	ID		**Parameter**
	T		0.0 to 127.15
	C		0.0 to 127.15
	D		0.0 to 255.15
	RS		0.0 to 255.15

Table 8-13 shows how the RLO is formed during the bit test operations “TB” and “TBN”. An example for applying the bit operations follows the table.

Table 8-13. Effect of “TB” and “TBN” on the RLO

Operation	TB		TBN	
Signal state of the bit in the operand indicated	0	1	0	1
Result of logic operation	0	1	1	0

Example	STL	Explanation
A photoelectric barrier that counts piece goods is installed at input I 0.0. After every 100 pieces, the program is to jump to FB5 or FB6. After 800 pieces, counter 10 is to be reset automatically and start counting again.	C DB 10 A I 0.0 CU C 10 A I 0.1 L KC 000 S C 10 O I 0.2 O F 5.2 R C 10 LD C 10 T DW 12 **TBN D 12.8** JC FB 5 **TB D 12.8** JC FB 6 **TB D 12.11** = F 5.2	Call data block 10. Input I 0.1 loads the count of counter 10 with the constant 0. With each positive edge change at I 0.0, the counter is incremented by 1. The counter is reset by either input I 0.2 or flag F 5.2. The current count of the counter is stored in data word DW12 in BCD code. As long as bit 8 of data word DW12 is zero, program processing jumps to function block FB5. This is the case for the first, third, fifth etc. batch of 100 pieces. As long as bit 8 of data word DW12 is "1", program scanning jumps to function block FB 6. This is the case for the second, fourth, sixth, etc. batch of 100 pieces. When data bit 11 of data word DW12 becomes "1" (the count is then 800), flag F 5.2 is set conditionally.
A photoelectric barrier that counts piece goods is installed at input I 0.3. After every 256 pieces, the counter is supposed to be reset and start counting again.	:A I 0.3 :CU C 2 :A I 0.4 :L KC 000 :S C 20 **:TB C 20.8** :JC = FULL :BEU **FULL:RU C 20.8** :BE	Input I 0.4 loads the count of counter 20 with the constant 0. The count is incremented by 1 with each positive edge change at input I 0.3. If the count has reached 256 = 100_H (bit 8 is "1"), program scanning jumps to the label "FULL". Otherwise the block is terminated.. Bit 8 of counter C 20 is set to "0" unconditionally. Then the count is again 000_H.

Note

Times and counts are stored in the timer/counter word in hexadecimal notation in the 10 least significant bits (bits 0 to 9).
The time base is stored in bits 12 and 13 of the timer word.

8.2.4 Digital Logic Operations

Digital logic operations combine the contents of both accumulators logically bit by bit.

Table 8-14 provides an overview of these digital logic operations. Examples follow the table.

Table 8-14. Overview of Digital Logic Operations

Operation	Operand	Meaning
AW		**Combine bit by bit through logic AND**
OW		**Combine bit by bit through logic OR**
XOW		**Combine bit by bit through logic EXCLUSIVE OR**

Processing a Digital Logic Operation

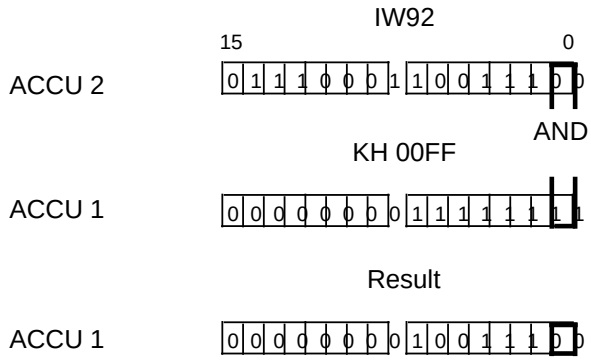
A digital logic operation is executed regardless of the RLO. It also does not affect the RLO. However, it sets condition codes according to the result of the arithmetic operation (see section 8.4).

> **Note**
>
> Make sure both operands have the same number format. Then load them into the accumulators before executing the operation.

The result of the arithmetic operation is available in ACCU 1 for further processing. The contents of ACCU 2 are not affected.

STL		Explanation
L	IW 92	Load input word IW92 into ACCU 1.
L	KH 00FF	Load a constant into ACCU 1. The previous contents of ACCU 1 are shifted to ACCU 2.
AW		Combine the contents of both accumulators bit by bit through logic AND.
T	QW 82	Transfer the resulting contents from ACCU 1 to output word QW82.

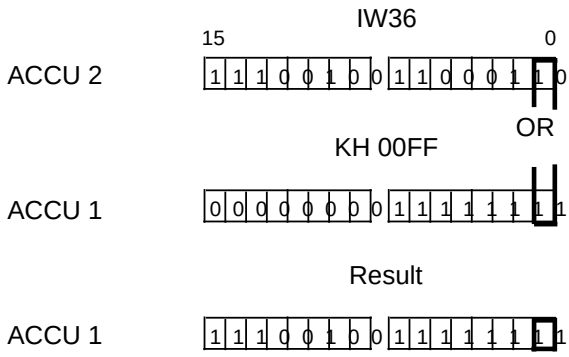
Numeric Example



Set the 8 high-order bits in input word IW92 to "0".
Compare both words bit by bit. If corresponding bits are both “1”, the result bit is set to "1".

STL		Explanation
L	IW 36	Load input word IW36 into ACCU 1.
L	KH 00FF	Load a constant into ACCU 1. The previous contents of ACCU 1 are shifted to ACCU 2.
OW		Combine the contents of both accumulators bit by bit through logic OR.
T	IW 36	Transfer the result (contents of ACCU 1) to input word IW36.

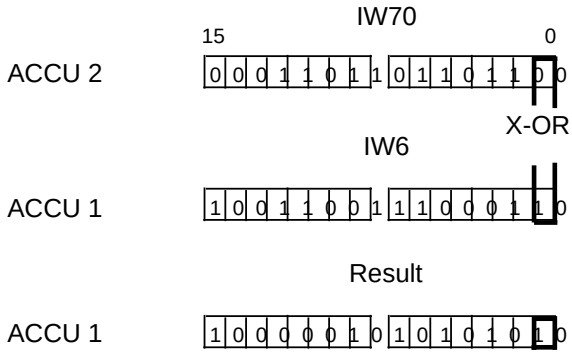
Numeric Example



Set the 8 low-order bits in input word IW36 to “1”. Compare both words bit by bit.
If either of the corresponding bits is “1”, a “1” is set in the result word.

STL		Explanation
L	IW 70	Load input word IW70 into ACCU 1.
L	IW 6	Load input word IW6 into ACCU 1. The previous contents of ACCU 1 are shifted to ACCU 2.
XOW		Combine the contents of both accumulators bit by bit through logic EXCLUSIVE OR.
T	QW 86	Transfer the result (contents of ACCU 1) to output word QW86.

Numeric Example



Check to see if input words IW70 and IW6 are equal.
The result bit is set to “1” only if corresponding bits in ACCU 1 and ACCU 2 are unequal.

8.2.5 Shift Operations

Shift operations shift a bit pattern in ACCU 1. The contents of ACCU 2 are not affected. Shifting multiplies or divides the contents of ACCU 1 by powers of two. Table 8-15 provides an overview of the shift operations. Examples follow the table.

Table 8-15. Overview of Shift Operations

Operation	Operand		Meaning
SLW			**Shift to the left.** The bit pattern in ACCU 1 is shifted to the left.
SRW			**Shift to the right.** The bit pattern in ACCU 1 is shifted to the right.
		↑ **Parameter**	0 to 15

Processing a Shift Operation

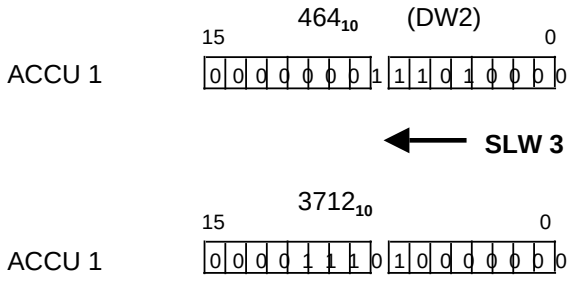
Execution of shift operations is unconditional. The RLO is not affected. However, shift operations set condition codes.
Consequently, the status of the last bit that is shifted out can be scanned with jump functions.

The shift statement parameter indicates the number of bit positions by which the contents of ACCU 1 are to be shifted to the left (SLW) or to the right (SRW). Bit positions vacated during shifting are assigned zeros.
The contents of the bits that are shifted out of ACCU 1 are lost. Following execution of the operation, the state of bit 2^0 (SRW) or bit 2^{15} (SLW) has an influence on the CC1 bit, which can then be evaluated.

A shift operation with parameter “0” is handled like a “NOP” operation. The central processor processes the next STEP 5 statement with no further reaction.
Before executing a shift operation, load the operand to be processed into ACCU 1.
The altered operand is available there for further processing.

STL	Explanation
L DW 2	Load the contents of data word DW2 into ACCU 1.
SLW 3	Shift the bit pattern in ACCU 1 three positions to the left.
T DW 3	Transfer the result (contents of ACCU 1) to data word DW3.

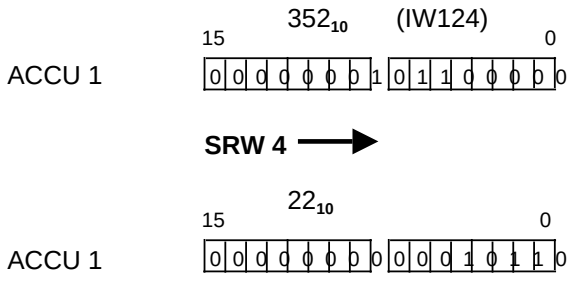
Numeric Example



The value 464_{10} is stored in data word DW2. Multiply this value by 2^3=8. Do so by shifting the bit pattern of DW2 in ACCU 1 three positions to the left.

STL	Explanation
L IW 124	Load the value of input word IW124 into ACCU 1.
SRW 4	Shift the bit pattern in ACCU 1 four positions to the right.
T QW 126	Transfer the result (contents of ACCU 1) to output word QW126.

Numeric Example



The value 352_{10} is stored in IW124. Shift the corresponding bit pattern in ACCU 1 four positions to the right to divide the value 352_{10} by 2^4 = 16.

8.2.6 Conversion Operations

Conversion operations convert the values in ACCU 1. Table 8-16 provides an overview of the conversion operations. Examples follow the table.

Table 8-16. Overview of Conversion Operations

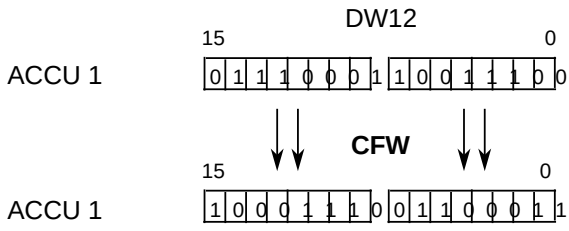
Operation	Operand	Meaning
CFW		**One's complement** The contents of ACCU 1 are inverted bit by bit.
CSW		**Two's complement** The contents of ACCU 1 are inverted bit by bit. Afterwards the word 0001_H is added.

Processing Conversion Operations

Execution of these operations does not depend on the RLO nor does it affect the RLO.
The “CSW” operation sets the condition codes (see section 8.4).

STL	Explanation
`L DW 12`	Load the contents of data word DW12 into ACCU 1.
`CFW`	Invert all bits in ACCU 1.
`T QW 20`	Transfer the new contents of ACCU 1 to output word QW20.

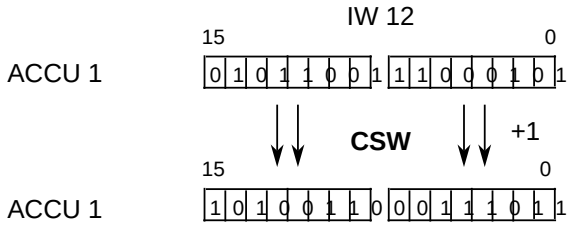
Numeric Example



In a system, normally open contacts have been replaced by normally closed contacts. If the information in data word DW12 is to maintain its previous effect, DW12 must be inverted.

STL	Explanation
L IW 12	Load the contents of input word IW12 into ACCU 1.
CSW	Invert all bits and add a “1”.
T DW 100	Transfer the altered word to data word DW100.

Numeric Example



Form the negative value of the value in input word IW12.

8.2.7 Decrement/Increment

The decrement/increment operations change the data loaded into ACCU 1. Table 8-17 provides an overview of the decrement/increment operations. An example follows the table.

Table 8-17. Decrement/Increment Operations

Operation	Operand		Meaning
D			**Decrement** Decrement the contents of the accumulator.
I			**Increment** Increment the contents of the accumulator. The contents of ACCU 1 are either decremented or incremented by the number indicated in the parameter. Execution of the operation is unconditional and is limited to the right-hand byte (without carry).
		↑	**Parameter** 0 to 255

Processing

Execution of the decrement and increment operations is independent of the RLO and does not affect the RLO or the condition codes.
The parameter indicates the value by which the contents of ACCU 1 are to be changed.
The operations refer to decimal values; however, the result is stored in ACCU 1 in binary form.
Changes relate only to the low byte in the accumulator.

Example	STL			Explanation
Increment the hexadecimal constant 1010_H by 16 and store the result in data word DW8.	C	DB	6	Call data block DB6.
	L	KH	1010	Load hexadecimal constant 1010_H into ACCU 1.
	I	16		Increment the low byte of ACCU 1 by 16. The result, 1020_H, is located in ACCU 1.
In addition, decrement the incrementation result by 33 and store the new result in data word DW9.	T	DW	8	Transfer the contents of ACCU 1 (1020_H) to data word DW8. Since the incrementation result is still in ACCU 1, you can decrement by 33 directly.
	D	33		The result would be FFF_H. However, since the high byte of ACCU 1 is not decremented along with the low byte, the result in ACCU 1 is $10FF_H$.
	T	DW	9	The contents of ACCU 1 are transferred to DW9 ($10FF_H$).

8.2.8 Disabling/Enabling Interrupts

The disable/enable interrupt operations affect interrupt-driven and time-controlled program scanning. They prevent process or time interrupts from interfering with the processing of a sequence of statements or blocks.

Note that the STATUS function cannot be used on blocks which are invoked between an IA and an RA operation in the quasi-safety mode.

Safety Note

Note that the interrupt response time may increase by the duration of the interrupt disable. If possible, never disable interrupts for more than 5 ms. If they are disabled for a longer period, the S5-95F responds as follows:

- After 5 ms, the S5-95F reports error 12, i.e. "Error in user program, interrupts disabled too long". There is no other response (the system remains at RUN).
- After 10 ms, the S5-95F resports error 4, i.e. "Too many interrupts", and goes to STOP to protect its failsafety.

If your process requires a system reponse after 5 ms, you must program it yourself. Single out the error with the number 13 by evaluating the system event DB in OB37.

Table 8-18. Disable/Enable Interrupt Operations

Operation	Operand		Meaning
IA			**Disable interrupt**
RA			**Enable interrupt**

Processing

Execution of the disable/enable interrupt operations does not depend on the RLO. These operations do not affect the RLO or the condition codes. After the “IA” statement is processed, no more interrupts are executed. The “RA” statement cancels the effect of “IA”.

Example	STL			Explanation
Disable interrupt processing in a specific program section and then enable it again.		.		
		.		
	=	Q	1.0	
	IA			Disable interrupt.
	A	I	0.0	
		.		
		.		
		.		
	JU	FB	3	If an interrupt occurs, the program section between the “IA” and “RA” is scanned without interruption.
		.		
		.		
		.		
	RA			Enable interrupt.
		.		Interrupts that occurred in the meantime are processed after the “RA” operation.
		.		
		.		

8.2.9 "DO" Operation

Use the "DO" operation to process STEP 5 statements as indexed operations. This allows you to change the parameter of an operand during control program processing (see Table 8-19).

Table 8-19. Overview of the "DO" Operation

Operation	Operand		Meaning
DO			**Processing a flag word or data word**
	ID ↑ FW DW	↑ **Parameter** 0 to 254 0 to 255	

"DO" Statements

"DO flag word or data word x" is a two-word statement that is unaffected by the RLO. "DO" consists of the following two statements:

- The first statement contains the "DO" operation and a flag word or data word.
- The second statement defines the operation and the operand identifier you want the control program to process. **You must enter 0 or 0.0 as the parameter**.

The control program works with the parameter that is stored in the flag word or data word. This parameter is the one called up in the first statement. If you want to index binary operations, inputs, outputs, or flags, you input the bit address in the high byte of this word. You input the byte address in the low byte.
In any other instance, the high byte must be "0".

The following operations can be combined with the "DO" statement:

Operations	Explanations
`L FY, T FY, L FW, T FW, L IB, T IB, L QB, T QB, L IW, T IW, L QW, T QW, L DL, T DL, L DR, T DR, L DW, T DW`	Load and transfer operations*
`CDB, JU OB, JU SB, JU FB, JU PB, JC OB, JC SB, JC FB, JC PB`	Block call operations (FBs must have no formal parameters)
`A F, = F, A T, AN T`	Boolean logic operations
`S F, R F, R T`	Set/reset operations
`SS T, SE T`	Timer operations
`SLW, SRW`	Shift operations

* Operations L IW0, T IW0, L QW0 and T QW0 are not permitted when slot 0 is configured with a redundant digital module.



Warning

Operations other than that listed in the above table are not permissible and will be rejected by the operating system.

Figure 8-6 shows how the contents of a data word determine the parameter of the next statement.

	DB6
DW 12	KH=**0108**
DW 13	KH=0001

FB x

```
:C    DB    6
      .
      .
      .
:DO   DW    12
:A    F     0.0
:DO   DW    13
:FR   T     0
```

Actual program

```
:C    DB    6
      .
      .
      .
      .
:A    F     8.1
      .
:FR   T     1
```

Figure 8-6. Executing a “DO” Operation

The following example illustrates how new parameters are generated in every program scan.

Example	STL	Explanation
Set the contents of data words DW20 to DW100 to signal state “0”. The index register for the parameter for the data words is DW1.	`:C DB 202`	Call data block DB202.
	`:L KB 20`	Load constant number 20 in ACCU 1.
	`:T DW 1`	Transfer contents from ACCU 1 to data word DW1.
	`F 1 :L KH 0`	Load hex constant 0 in ACCU 1.
	`:DO DW 1`	DO data word DW1.
	`:T DW 0`	Transfer the contents from ACCU 1 to the data word whose address is stored in data word DW1.
	`:L DW 1`	Load data word DW1 in ACCU 1.
	`:L KB 1`	Load constant number 1 in ACCU 1. Data word DW1 is shifted to ACCU 2.
	`:+F`	ACCU 2 und ACCU 1 are added, and the result is stored in ACCU 1 (data word address is higher).
	`:T DW 1`	Transfer contents of ACCU 1 to data word DW1 (new data word address).
	`:L KB 100`	The constant number 100 is loaded in ACCU 1 and the new data word address is shifted to ACCU 2.
	`:<=F`	Compare the ACCUs for less than or equal to: ACCU 2 ACCU 1.
	`:JC = F 1`	Jump conditionally to label F1, if ACCU 2 ACCU 1.

8.2.10 Jump Operations

Table 8-20 provides an overview of the jump operations. An example follows the table.

Table 8-20. Overview of Jump Operations

Operation	Operand		Meaning
JU =			**Jump unconditionally** The unconditional jump is executed independently of conditions.
JC=			**Jump conditionally** The conditional jump is executed if the RLO is “1”. If the RLO is “0”, the statement is not executed and the RLO is set to “1”.
JZ =			**Jump if the result is “zero”** The jump is executed only if CC 1 = 0 and CC 0 = 0 The RLO is not changed.
JN =			**Jump if the result is “not zero”** The jump is executed only if CC 1 CC 0 The RLO is not changed.
JP =			**Jump if the result is positive** The jump is executed only if CC 1 = 1 and CC 0 = 0 The RLO is not changed.
JM =			**Jump if the result is negative** The jump is executed only if CC 1 = 0 and CC 0 = 1 The RLO is not changed.
JO =			**Jump on overflow** The jump is executed if an overflow occurs. Otherwise the jump is not executed. The RLO is not changed.
ID Jump label (up to 4 characters)			

Processing Jump Operations

A symbolic jump destination (jump label) must always be entered next to a jump operation. This jump label can have up to four characters. The first character must be a letter of the alphabet.

When programming, please be aware of the following items:

- The absolute jump displacement cannot exceed +127 or - 128 words in the program memory. Some statements take up two words (e.g., "Load a constant"). For long jumps, insert an intermediate destination.
- Jumps can be executed only within a block.
- Jumping over segment boundaries ("BLD 255") is not permitted.
- Jump labels can be set only if the preceding operation has reloaded the RLO. Jump labels to a sequence of binary scans (RLO sequence) are recognized by the operating system and are rejected with an error message.
- A jump to a label marking statement **A(** or **O(** is not permitted. The S5-95F's compiler recognizes an illegal statement and the S5-95F responds by going to STOP. In this case, program label and statement on two different lines.

Example	STL		Explanation
If no bit of input word IW1 is set, program scanning jumps to the label "AN 1". If input word IW1 and output word QW3 do not agree, program processing jumps back to the label "AN 0". Otherwise input word IW1 and data word DW12 are compared. If input word IW1 is greater than or less than data word DW12, program scanning jumps to the "DEST" label.	AN0	:L IW 1	Load input word IW1 into ACCU 1. If the contents of ACCU 1 equal zero[1], jump to the label "AN 1". Otherwise process the next statement (I 0.0).
		:L KH 0000	
		:+F	
		:JZ= AN 1	
		:A I 0.0	
		.	
		.	
		.	
		.	
		.	
		:= Q4.0	End of an RLO sequence
	AN1	:L IW 1	Compare input word IW1 and output word QW3. If they are not equal, set individual bits in ACCU 1.
		:L QW 3	
		:XOW	
		:JN = AN 0	If the contents of ACCU 1 are not zero, jump to the label "AN 0". Otherwise process the next statements.
		:L IW 1	Compare input word IW1 and data word DW12. If they are not equal, set RLO to "1".
		:L DW12	
		:>< F	
		:JC = DEST	If the RLO = "1", jump to the "DEST" label. If the RLO = "0", process the next statement.
		.	
		.	
		.	
		.	
		.	
	DEST	:A I 0.1	
		.	
		.	

1 The "L..." statement does not affect the condition codes. An addition (+F) is executed with the constant 0000_H so that the "JZ" operation can evaluate the contents of the accumulator.

8.2.11 Substitution Operations

If you plan to process a program with various operands and without a lot of changes, it is advisable to assign parameters to individual operands (see section 7.3.4).

If you have to change the operands, you only need to reassign the parameters in the function block call.

These parameters are processed in the program as "formal oparands".

Special operations are necessary for this processing. However, these special operations are no different in their effect than operations without substitution. A brief description of these operations and examples follows.

Binary Logic Operations

Table 8-21 provides an overview of binary logic operations.

Table 8-21. Overview of Binary Logic Operations

Operation	Operand	Meaning
A =		**AND operation** Scan a formal operand for "1".
AN =		**AND operation** Scan a formal operand for "0".
O =		**OR operation** Scan a formal operand for "1".
ON =		**OR operation** Scan a formal operand for "0".

Formal operand	Actual Operands Permitted	Parameter Type	Data Type
	Inputs, outputs, and flags addressed in binary form	I , Q, F	BI
	Timers and counters	T , C	

Set/Reset Operations

Table 8-22 provides an overview of the set/reset operations. An example follows the table.

Table 8-22. Overview of Set/Reset Operations

Operation	Operand	Meaning
S =		**Set a formal operand (binary).**
RB =		**Reset a formal operand (binary).**
= =		**Assign** The RLO is assigned to a formal operand.

Formal operand	Actual Operands Permitted	Parameter Type	Data Type
	Inputs, outputs, and flags addressed in binary form	I , Q, F	BI

Example: FB30 is assigned parameters in OB1.

Call in OB1		Program in FB30		Executed Program		
	:JU FB 30	**:A**	**=ON1**	:A	I	0.0
NAME	:COMBINE	**:AN**	**=ON2**	:AN	I	0.1
ON 1	: I 0.0	**:O**	**=ON3**	:O	I	0.2
ON 2	: I 0.1	**:S**	**=MOT5**	:S	Q	1.2
ON 3	: I 0.2	**:=**	**=OFF1**	:=	Q	1.0
VAL1	: I 0.3	**:A**	**=VAL1**	:A	I	0.3
OFF1	: Q 1.0	**:A**	**=ON2**	:A	I	0.1
OFF2	: Q 1.1	**:ON**	**=ON3**	:ON	I	0.2
MOT5	: Q 1.2	**:RB**	**=MOT5**	:R	Q	1.2
	: BE	**:=**	**=OFF2**	:=	Q	1.1
		:BE		:BE		

Load and Transfer Operations

Table 8-23 lists the various load and transfer operations. An example follows the table.

Table 8-23. Overview of Load and Transfer Operations

Operation	Operand	Meaning
L =		**Load a formal operand.**
LD =		**Load a formal operand in BCD code.**
LW =		**Load the bit pattern of a formal operand.**
T =		**Transfer to a formal operand.**

Formal operand	Actual Operands Permitted	Parameter Type	Data Type
For L =	Inputs, outputs, and flags addressed in binary form Data Timers and counters	I, Q, F PW*, PY* DW, DR, DL T, C	BY, W
For LD =	Timers and counters	T, C	
For LW =	Bit pattern	D	KF, KH, KM, KY, KS, KT, KC
For T =	Inputs, outputs, data (DW, DR, DL) and flags addressed in binary form	I, Q DW, DR, DL F, PW*, PY*	BY, W

* Not for integral function blocks

Example: FB34 is assigned parameters in PB1.

Call in PB1

```
        :JU FB 34
NAME    :LOAD/TRAN
I0      :   I 0.0
I1      :   I 0.1
L1      :   FW 10
LW1 :   :   KC 140
LC1     :   C 7
T1      :   QW 4
LW2 :   :   KC 160
        :BE
```

Program in FB34

```
:A      =I 0
:L      =L1
:S      C       6
:A      =I 1
:LW     =LW1
:S      C       7
:A      I       0.2
:CU     C       6
:CU     C       7
:LD     =LC1
:T      =T1
:A      I       0.3
:R      C       6
:R      C       7
:LW     =LW2
:LD     =LC1
:!=F
:R      C       7
:BE
```

Executed Program

```
:A      I       0.0
:L      FW      10
:S      C       6
:A      I       0.1
:L      KC      140
:S      C       7
:A      I       0.2
:CU     C       6
:CU     C       7
:LD     C       7
:T      QW      4
:A      I       0.3
:R      C       6
:R      C       7
:L      KC      160
:LD     C       7
:!=F
:R      C       7
:BE
```

Timer and Counter Operations

Table 8-24 provides an overview of timer and counter operations. Examples follow the table.

Table 8-24. Overview of Timer and Counter Operations

Operation	Operand	Meaning
FR =		**Enable** a formal operand for a cold restart. (For a description, see “FT” or “FC”, according to the formal operand).
RD =		**Reset** a formal operand (digital).
SP =		**Start** a pulse timer specified as a formal operand using the value stored in the accumulator.
SD =		**Start** an on-delay timer specified as a formal operand using the value stored in the accumulator.
SEC =		**Start** an extended pulse timer specified as a formal operand using the value stored in the accumulator or **set** a counter specified as a formal operand using the count specified in the accumulator.
SSU =		**Start** a stored on-delay timer specified as a formal operand using the value stored in the accumulator or start the **count up** of a counter specified as a formal operand.
SFD =		**Start** an off-delay timer specified as a formal operand using the value stored in the accumulator or start the **count down** of a counter specified as a formal operand.

Formal operand	Actual Operands Permitted	Parameter Type	Data Type
	Timers and counters[1]	T, C[1]	

[1] “SP” and “SD” do not apply to counters.

Specifying Times and Counts

As with the basic operations, you can specify a time or count as a formal operand. In this case, you must distinguish as follows whether the value is located in an operand word or is specified as a constant.

- Operand words can be of parameter type "I" or "Q" and of data type "W". Use the "L=" operation to load them into the accumulator.
- Constants can be of parameter type "D" and of data type "KT" or "KC". Use "LW=" to load these formal operands into the accumulator

The following examples show how to work with timer and counter operations:

Example 1:

Function Block Call	Program in Function Block (FB32)	Executed Program
:JU FB 32 NAME :TIME I 5 : I 0.0 I 6 : I 0.1 TIM5 : T 5 TIM6 : T 6 OFF6 : Q 1.0 :BE	:AN =I 5 :A =I 6 :L KT 005.2 **:SFD =TIM5** :A =I 5 :AN =I 6 :L KT 005.2 **:SSU =TIM6** :A =TIM5 :O =TIM6 := =OFF6 :A I 0.2 **:RD =TIM5** **:RD =TIM6** **:BE**	:AN I 0.0 :A I 0.1 :L KT 5.2 :SF T 5 :A I 0.0 :AN I 0.1 :L KT 5.2 :SS T 6 :A T 5 :O T 6 := Q 1.0 :A I 0.2 :R T 5 :R T 6 :BE

Example 2:

Function Block Call	Program in Function Block (FB33)	Executed Program
:JU FB 33 NAME :COUNT I2 : I 0.0 I3 : I 0.1 I4 : I 0.2 CNT5 : C 5 OFF3 : Q 1.0 :BE	:A =I 2 :L KC 017 **:SEC =CNT5** :A =I 3 **:SSU =CNT5** :A =I 4 **:SFD =CNT5** :A =CNT5 := =OFF3 :A I 0.3 **:RD =CNT5** :BE	:A I 0.0 :L KC 017 :S C 5 :A I 0.1 :CU C 5 :A I 0.2 :CD C 5 :A C 5 := Q 1.0 :A I 0.3 :R C 5 :BE

"DO" Operation

Table 8-25 and the example that follows explain the processing operation.

Table 8-25. "DO" Operation

Operation	Operand	Meaning
DO =		**Process formal operand** The substituted blocks are called unconditionally.

Formal operands	Actual Operands Permitted	Parameter Type	Data Type
	DB, PB, SB, FB[1]	B	

[1] As actual operands, function blocks cannot have block parameters.

> **Note**
>
> This command is in any case reloading the RLO independent of the command actually executed.

Example:

Function Block Call	Program in Function Block FB35	Executed Program
STL :JU FB 35 NAME :DO D5 : DB 5 DW2 : DW 2 D6 : DB 6 DW1 : DW 1 Q4 : QW 4 MOT5 : FB 36 :BE	**:DO =D5** :L =DW2 **:DO =D6** :T =DW1 :T =Q4 **:DO =MOT5** :BE	:C DB 5 :L DW 2 :C DB 6 :T DW 1 :T QW 4 :JU FB 36 :BE

8.3 System Operations

System operations and supplementary operations have the following limitations:

- You can program them only in function blocks.
- You can program them only in the STL method of representation.

Since system operations access system data, only users with very profound system knowledge should use them.

8.3.1 Set Operations

Like the supplementary bit operations, these set operations can change individual bits. Table 8-26 provides an overview of the set operations.

Table 8-26. Overview of Set Operations

Operation	Operand		Meaning
SU			**Set bit unconditionally** A specific bit is set to "1" in the system data area.
RU			**Reset bit unconditionally** A specific bit is set to "0" in the system data area.
	ID RS		**Parameter** 0.0 to 255.15

Processing Set Operations

Execution of set operations does not depend on the RLO.

8.3.2 Load and Transfer Operations

Use these load and transfer operations to address the entire program memory of the programmable controller. They are used mainly for data exchange between the accumulator and memory locations that cannot be addressed by operands. Table 8-27 provides an overview of the load and transfer operations.

Table 8-27. Overview of Load and Transfer Operations

Operation	Operand	Meaning
LIR		**Load the register indirectly** The contents of a memory word are loaded into the specified register (ACCU 1, 2). The address is in ACCU 1.
TIR		**Transfer the register indirectly** The contents of the indicated register are transferred to a memory location. The address is in ACCU 1.
	Parameter 0 (for ACCU 1), 2 (for ACCU 2)	
TNB		**Transfer a data field (byte-by-byte)** A memory area is transferred in the program memory as a field. End address destination area: ACCU 1 End address source area: ACCU 2
T		**Transfer** A word is transferred to the system data area.
ID RS	**Parameter** 0 to 255	

Loading and Transferring Register Contents

Both accumulators can be addressed as registers. Each register is 16 bits wide. Since the "LIR" and "TIR" operations transmit data by words, the registers are addressed in pairs.
Loading and transferring register contents are independent of the RLO. The processor goes to ACCU 1 to get the address of the memory location referenced during data exchange. Consequently, make sure that the desired address is stored in ACCU 1 before this system operation is processed.

STL			Explanation
	.		
	.		
L	KH	7E10	Load the address $7E10_H$ into ACCU 1.
LIR	0		Load the information from the memory location with the address $7E10_H$ into ACCU 1.

Restrictions for the LIR, TIR and TNB Operations

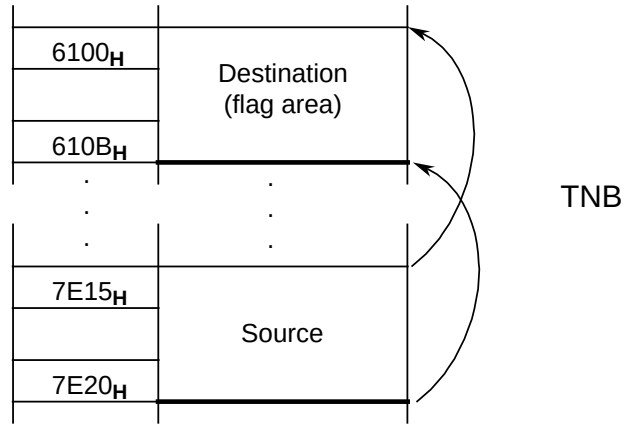
Command	Restriction
LIR	The following address areas must not be accessed: $4F00_H$ ... $4FFF_H$ 5900_H ... $5C7F_H$ 6400_H ... $65FF_H$ 7000_H ... $73FF_H$ 8000_H ... $FFFF_H$
TIR	Access permitted only to flag area and open data blocks
TNB	For all source bytes: same restrictions as LIR For all destination bytes: same restriction as TIR

The S5-95F goes to STOP if the illegal address areas are accessed.

Processing a Field Transfer

A field transfer is processed independently of the RLO.
The parameter indicates the length of the data field (in bytes) that is to be transferred. The field can be up to 255 bytes long.

The address of the source field is in ACCU 2. The address of the destination field is in ACCU 1.
The higher address of each field must be specified because a field transfer takes place by decrementing. The bytes in the destination field are overwritten during the transfer.

Example	Representation
Transfer a 12-byte data field (address $7E20_H$ to $7E15_H$) to flag area FY11 ... FY0 (address from $610B_H$ to 6100_H).	

STL	Explanation
:L KH 7E20	The end address of the source field is loaded into ACCU 1.
:L KH 610B	The end address of the destination field is loaded into ACCU 1. The source address is shifted to ACCU 2.
:TNB 12	The data field is transferred to the destination field.

Caution

The TIR, TRS and TNB operations are memory-changing operations with which you can access the user memory and/or the system data area. Improper use of the operations can lead to changes in the control program and to malfunctioning of the S5-95F.

8.3.3 Arithmetic Operations

An arithmetic operation changes the contents of ACCU 1 by a specified value. The parameter represents this value as a positive or negative decimal number.

Table 8-28. Arithmetic Operation

Operation	Operand		Meaning
ADD			**Add a constant** Add byte or word constants.
	ID	**Parameter**	
	BF	-128 to +127	
	KF	-32768 to +32767	

Processing

An arithmetic operation is executed independently of the RLO. It does not affect the RLO or the condition codes.

You can subtract by entering a negative parameter.

Even if the result cannot be represented by 16 bits, no carry is made to ACCU 2, i.e., the contents of ACCU 2 are not changed.

Example	STL	Explanation
Decrement the constant 1020_H by 33 and store the result in flag word FW28. Afterwards add the constant 256 to the result and store the sum in flag word FW30.	`L KH 1020` `ADD BF -33` `T FW 28` `ADD KF 256` `T FW 30`	The constant 1020_H is loaded into ACCU 1. The constant -33_{0D} is added to the ACCU contents. The new ACCU contents ($0FFF_H$) are stored in flag word FW28. The constant 256_{0D} is added to the last result. The new ACCU contents ($10FF_H$) are stored in flag word FW30.

8.3.4 Other Operations

Table 8-29 provides an overview of the remaining system operations.

Table 8-29. The "TAK" and "STS" Operations

Operation	Operand	Meaning
TAK		**Swap accumulator contents** Swap the contents of ACCU 1 and ACCU 2 regardless of the RLO. The RLO and the condition codes are not affected.
STS		**Stop immediately** The PLC goes into the STOP mode regardless of the RLO.

Processing the "STS" Operation

When the "STS" operation is executed, the programmable controller goes into the STOP mode immediately. Program processing is terminated at this point.

To return the PLC to RUN mode:

- In safety mode you can only do this manually using the RUN/STOP switch
- In test mode you can do this either manually using the RUN/STOP switch or on a programmer using the programmer function ”PLC Start”.

Note

In safety mode a STOP initiated by the ***”STS” operation*** can only be cancelled manually by operating the two RUN/STOP switches. A STOP initiated by the ”STS” operation is regarded as a ***safety stop*** in safety mode.
A STOP initiated by the ”STS” operation has the same status in safety mode as a STOP produced by the RUN/STOP switches.

8.4 Condition Code Generation

The processor of the programmable controller has the following three condition codes:

- CC 0
- CC 1
- OV (overflow)

The following operations affect the condition codes.

- Comparison operations
- Arithmetic operations
- Shift operations
- Some conversion operations

The state of the condition codes represents a condition for the various jump operations.

Condition Code Generation for Comparison Operations

Execution of comparison operations sets condition codes CC 0 and CC 1 (see Table 8-30). The overflow condition code is not affected. Comparison operations do affect the RLO. When a comparison is satisfied, the RLO is 1. This allows you to use the “JC” conditional jump operation after a comparison operation.

Table 8-30. Condition Code Settings for Comparison Operations

Contents of ACCU 2 as Compared to Contents of ACCU 1	Condition Codes			Possible Jump Operations
	CC 1	CC 0	OV	
Equal to	0	0		JZ
Less than	0	1		JN, JM
Greater than	1	0		JN, JP

Condition Code Generation for Arithmetic Operations

Execution of arithmetic operations sets all condition codes according to the result of the arithmetic operation (see Table 8-31).

Table 8-31. Condition Code Settings for Fixed-Point Arithmetic Operations

Result after Arithmetic Operation is Executed	Condition Codes			Possible Jump Operations
	CC 1	CC 0	OV	
< - 32768	1	0	1	JN, JP, JO
- 32768 to - 1	0	1	0	JN, JM
0	0	0	0	JZ
+1 to +32767	1	0	0	JN, JP
> +32767	0	1	1	JN, JM, JO
(-) 65536*	0	0	1	JZ, JO

* This number is the result of the calculation -32768 - 32768

Condition Code Generation for Digital Logic Operations

Digital logic operations set CC 0 and CC 1. They do not affect the overflow condition code (see Table 8-32). The setting depends on the contents of the ACCU after the operation has been processed.

Table 8-32. Condition Code Settings for Digital Logic Operations

Contents of the ACCU	Condition Codes			Possible Jump Operations
	CC 1	CC 0	OV	
Zero (KH = 0000)	0	0		JZ
Not zero	1	0		JN, JP

Condition Code Generation for Shift Operations

Execution of shift operations sets CC 0 and CC 1. It does not affect the overflow condition code (see Table 8-33).

Code setting depends on the state of the last bit shifted out.

Table 8-33. Condition Code Settings for Shift Operations

Value of the Last Bit Shifted Out	Condition Codes			Possible Jump Operations
	CC 1	CC 0	OV	
“0”	0	0		JZ
“1”	1	0		JN, JP

Condition Code Generation for Conversion Operations

The formation of the two's complement (CSW) sets all condition codes (see Table 8-34). The state of the condition codes is based on the result of the conversion function.

Table 8-34. Condition Code Settings for Conversion Operations

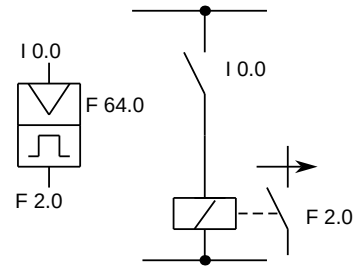
Result after Arithmetic Operation is Executed	Condition Codes			Possible Jump Operations
	CC 1	CC 0	OV	
- 32768 *	0	1	1	JN, JM, JO
- 32767 to - 1	0	1	0	JN, JM
0	0	0	0	JZ
+1 to +32767	1	0	0	JN, JP

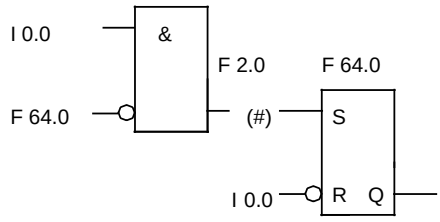
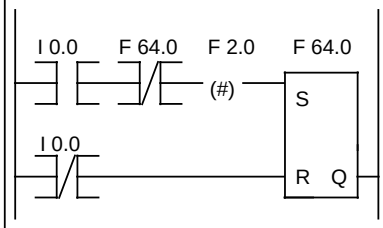
* This number is the result of the conversion of KH = 8000.

8.5 Sample Programs

Sections 8.5.1 through 8.5.3 provide a few sample programs that you can enter and test in all three methods of representation on a programmer.

8.5.1 Momentary-Contact Relay/Edge Evaluation

Example	Circuit Diagram
On each leading edge of the signal at input I 0.0, the AND condition “A I 0.0 and AN F 64.0” is satisfied; the RLO is “1”. This sets flags F 64.0 and F 2.0 (“edge flags”). In the next processing cycle, the AND condition “A I 0.0 and AN F 64.0” is not satisfied since flag F 64.0 has already been set. Flag 2.0 is reset. Therefore, flag F 2.0 is “1” for only one program run. When input I 0.0 is switched off, flag F 64.0 is reset. This resetting prepares the way for evaluation of the next leading edge of the signal at input I 0.0.	

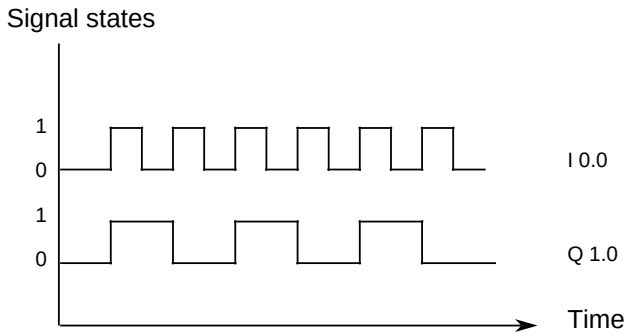
STL	CSF	LAD
A I 0.0 AN F 64.0 = F 2.0 S F 64.0 AN I I0.0 R F 64.0 NOP 0		

8.5.2 Binary Scaler/Binary Divider

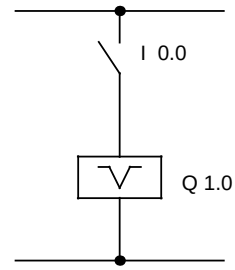
This section describes how to program a binary scaler.

Example: The binary scaler (output Q 1.0) changes its state each time I 0.0 changes its signal state from “0” to “1” (leading edge). Therefore, half the input frequency appears at the output of the flip-flop.

Timing Diagram



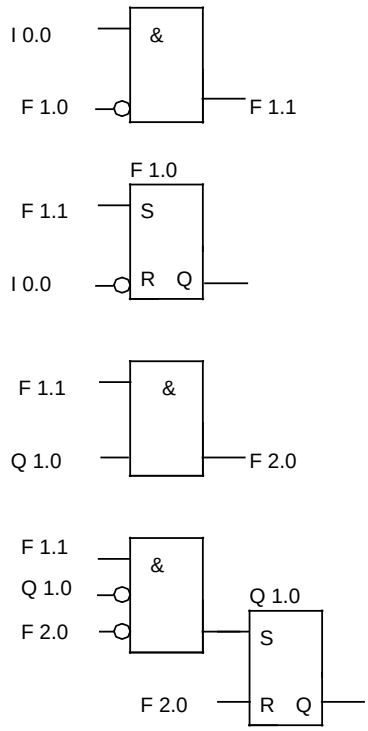
Circuit Diagram



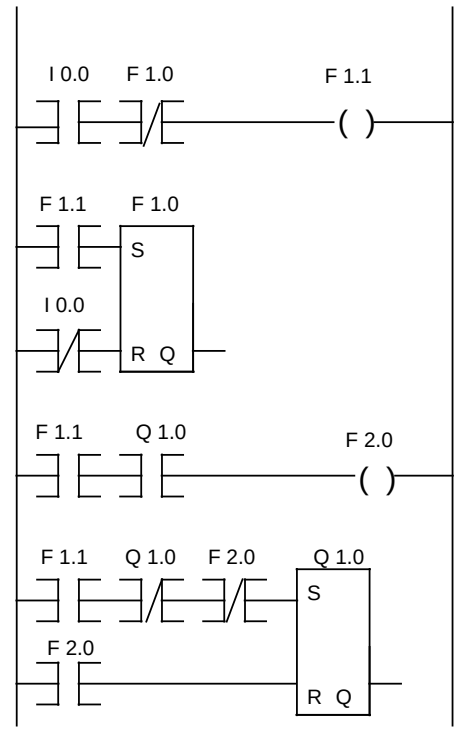
STL

```
A     I     0.0
AN    F     1.0
=     F     1.1
***
A     F     1.1
S     F     1.0
AN    I     0.0
R     F     1.0
NOP 0
***
A     F     1.1
A     Q     1.0
=     F     2.0
***
A     F     1.1
AN    Q     1.0
AN    F     2.0
S     Q     1.0
A     F     2.0
R     Q     1.0
NOP 0
***
```

CSF



LAD



Note

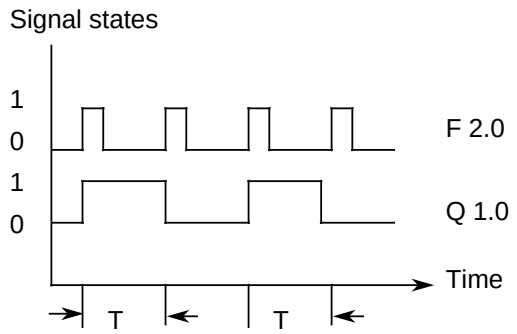
Output in CSF or LAD is possible only if you enter the segment boundaries "***" when programming in STL.

8.5.3 Clock/Clock-Pulse Generator

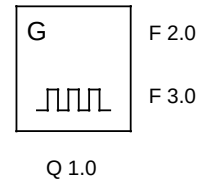
This subsection describes how to program a clock-pulse generator.

Example: A clock-pulse generator can be implemented using a self-clocking timer that is followed in the circuit by a binary scaler. Flag F 2.0 restarts timer T 7 each time it runs down, i.e., flag F 2.0 is "1" for one cycle each time the timer runs down. The pulses of flag F 2.0 applied to the binary scaler result in a pulse train with pulse duty factor 1:1 at output Q 1.0. The period of this pulse train is twice as long as the time value of the self-clocking timer.

Timing Diagram



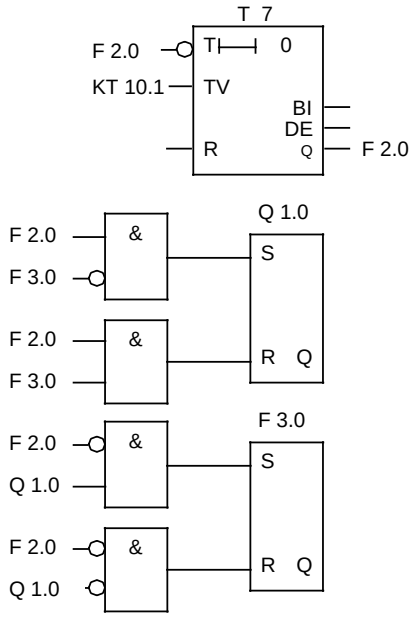
Circuit Diagram



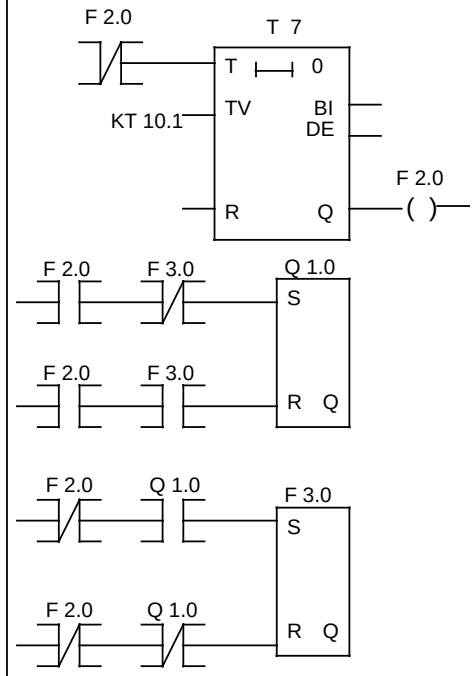
STL

```
AN    F     2.0
L     KT  010.1
SD    T     7
NOP   0
NOP   0
NOP   0
A     T     7
=     F     2.0
***
A     F     2.0
AN    F     3.0
S     Q     1.0
A     F     2.0
A     F     3.0
R     Q     1.0
NOP   0
***
AN    F     2.0
A     Q     1.0
S     F     3.0
AN    F     2.0
AN    Q     1.0
R     F     3.0
NOP   0
```

CSF



LAD



9 Blocks and Their Functions

9.1	Organization Blocks	9 - 1
9.1.1	Scan Time Trigger (OB31)	9 - 2
9.1.2	Procedure after Battery Failure (OB34)	9 - 2
9.1.3	Error Handling (OB37)	9 - 2
9.1.4	PID Algorithm (OB 251)	9 - 3
9.2	Integrated Function Blocks	9 - 14
9.2.1	2-out-of-3 Evaluation for Failsafe Digital Inputs -FB234-	9 - 15
9.2.2	Interfacing Operator Panels and Text Displays via the CP 521 SI's Serial Port -FB235-	9 - 23
9.2.3	Function Block FB236 for Non-Equivalence and Watchdog Timers	9 - 34
9.2.4	Function Block FB 237, Interpolation Block	9 - 41
9.2.5	Code Converter : B4 - FB240 -	9 - 53
9.2.6	Code Converter : 16 - FB241 -	9 - 53
9.2.7	Multiplier : 16 - FB242 -	9 - 54
9.2.8	Divider : 16 - FB243 -	9 - 54
9.2.9	Additional Calling Up of Test Routines - FB252 -	9 - 55
9.2.10	Depassivation Block - FB255 -	9 - 61
9.3	Parameterizing Internal Functions in DB1	9 - 63
9.3.1	Configuration and Default Settings for DB1	9 - 64
9.3.2	How to Assign Parameters in DB1 without COM 95F	9 - 66
9.3.3	Rules for Setting Parameters in DB1	9 - 66
9.3.4	How to Recognize and Correct Parameter Errors	9 - 68
9.3.5	Transferring Changed DB1 Parameters to the S5-95F Programmable Controller	9 - 69
9.3.6	Reference Guide for Setting Parameters in DB1	9 - 70

Figures

9-1 Calling Up the OB251 PID Algorithm . . . . . 9 - 3
9-2 Block Diagram of the PID Controller . . . . . 9 - 4
9-3 Principle of Interval Sampling . . . . . 9 - 9
9-4 Process Schematic . . . . . 9 - 10
9-5 The Status Word . . . . . 9 - 25
9-6 Example for Output of an Error Block on the OP15 . . . . . 9 - 32
9-7 Function Characteristic . . . . . 9 - 48
9-8 Value Table . . . . . 9 - 49
9-9 List of S5-95F Self-Test Components . . . . . 9 - 55
9-10 DB1 with Default Parameters . . . . . 9 - 64

Tables

9-1 Overview of Organization Blocks . . . . . 9 - 1
9-2 Restrictions of the STEP 5 Commands in Various Blocks . . . . . 9 - 1
9-3 Legend for the Block Diagram of the PID Controller (Figure 9-2) . . . . . 9 - 4
9-4 Description of the Control Bits in Control Word "STEU" . . . . . 9 - 5
9-5 Format of the Controller DB . . . . . 9 - 7
9-6 Block Parameters for FB234 . . . . . 9 - 16
9-7 Reserved Operands and Blocks . . . . . 9 - 17
9-8 Cycle Time Requirements . . . . . 9 - 17
9-9 Modes . . . . . 9 - 19
9-10 Errors Reported by the Operating System . . . . . 9 - 20
9-11 Overview of Possible Configurations for 2-out-of-3 Evaluation with FB234 . . 9 - 21
9-12 FB235 Block Parameters . . . . . 9 - 24
9-13 Error Codes in the Status Word . . . . . 9 - 26
9-14 FB235 Reserved Operands . . . . . 9 - 26
9-15 The Interface DB . . . . . 9 - 30
9-16 Example for Defining Field Texts . . . . . 9 - 30
9-17 Output of S5-95F Error Messages via Field Texts . . . . . 9 - 31
9-18 FB236 Block Parameters . . . . . 9 - 35
9-19 Reserved Operands . . . . . 9 - 35
9-20 Overview of Possible Configurations for FB236 (Non-Equivalence Tests) . . . . . 9 - 38
9-21 Overview of Possible Configurations for FB236 (Watchdog Timers) . . . . . 9 - 39
9-22 FB237 Block Parameters . . . . . 9 - 42
9-23 Additional Remarks Regarding Block Parameters . . . . . 9 - 43
9-24 Reserved FB237 Operands . . . . . 9 - 43
9-25 Data Blocks DBP1/DBP2 for the Rough Characteristic . . . . . 9 - 46
9-26 Data Blocks DBP1/DBP2 for Precise Characteristic Sections . . . . . 9 - 47
9-27 Calling and Parameter Assignments of FB240 . . . . . 9 - 53
9-28 Calling and Parameter Assignments of FB241 . . . . . 9 - 53
9-29 Calling and Parameter Assignments of FB242 . . . . . 9 - 54
9-30 Calling and Parameter Assignments of FB243 . . . . . 9 - 54
9-31 Calling and Parameter Assignments of FB252 . . . . . 9 - 57
9-32 Execution Times of FB252 . . . . . 9 - 59
9-33 Parameter Blocks and Their IDs (S5-95F) . . . . . 9 - 65

9 Blocks and Their Functions

The STEP 5 programming language offers you various blocks, such as organization blocks (OBs), program blocks (PBs), function blocks (FBs) and data blocks (DBs). Sequence blocks (SBs) are available for programming of sequencers.
The following chapter describes:

- the meaning of the organization blocks
- the blocks integrated in the S5-95F

9.1 Organization Blocks

Table 9.1 lists the organization blocks of the S5-95F.

Table 9-1. Overview of Organization Blocks

OB No.	Function
OB1	Cyclic program processing
OB2	Interrupt-driven program processing (hardware interrupt)
OB3	Interrupt-driven program processing (software interrupt ”Counter overflow’)
OB13	Time-controlled program processing
OB21	Restart procedure after manual POWER ON (STOP RUN)
OB22	Restart procedure after power recovery
OB31	Scan time triggering of hardware watchdog
OB34	Battery failure
OB37	Error handling
OB251	PID algorithm

Table 9-2. Restrictions of the STEP 5 Commands in Various Blocks

Command	OB 1 OB 34 OB 37	OB 2	OB 3	OB 13	OB 21 OB 22
Boolean logic operations	X	only with IB 32, 33, 56, 60	X	X	X[1)]
Set/reset operations	X	only RQ 32.x RI 60.x	X	X	X[1)]
Load and transfer operations	X[2)]	-	X[2)]	X	X[1)2)]
Timer, counter, comparison, block call operations	X	only BE	X	X	X
Supplementary operations	0	-	0	0	0
System operations	0	-	0	0	0

X = permissible in the OB and in PBs, SBs and FBs called there
0 = permissible only in called-up FBs
- = not permissible

1) Impractical with PII
2) Direct access to external I/Os not permissible

9.1.1 Scan Time Trigger (OB31)

By means of a scan time monitor (hardware watchdog) you monitor the time sequence of cyclic program processing. If program processing takes longer than the cycle monitoring time of 680 ms set via the hardware, then the S5-95F goes to STOP. This can occur, for example:

- if the control program is too long
- if you have programmed an infinite loop.

By calling the integral organization block OB31, you can retrigger the scan time monitor at any point in the control program, i.e. re-initiate the cycle monitoring time. OB31 is already programmed and you then only need to call it up. However, you cannot change this organization block.

Calling OB31:

At any point in the control program: Enter JU OB31.

> **Note**
>
> For safety reasons, the scan time is monitored not only by the hardware watchdog but also by a software watchdog. You specify the monitoring time for the software watchdog in DB 1; retriggering of this monitoring time in the control program is not possible.

9.1.2 Procedure after Battery Failure (OB34)

The S5-95F checks the status of the battery. In the case of a battery failure, the S5-95F reacts as follows:

- The battery failure is signalled
- A message is entered in the system event DB (see Table 15.2)
- After the battery failure you must replace the battery within 72 hours. During this time, the S5-95F calls up OB34 prior to each cycle.
- If the battery is not replaced within 72 hours, the S5-95F goes to STOP. Prior to this it enters the error occurred (battery failure) in the system event data block DB254 and calls up the error handling organization block OB37 once.

9.1.3 Error Handling (OB37)

In the organization block OB37 you can react to signalled errors and messages specifically. OB37 is called and processed by the operating system every time the S5-95F enters an error or a message in the system event DB (DB254). Every entry in the system event DB results in OB37 being called once (no cyclic calling of OB37).

9.1.4 PID Algorithm (OB 251)

A PID algorithm is integrated in the operating system of the S5-95F. OB251 helps you use this algorithm to meet your needs.

Before calling up OB251, you must first open a data block called the controller DB. It contains the controller parameters and other controller specific data. The PID algorithm must be invoked periodically and generates the manipulated variable. The more closely the scan time is maintained, the more accurately the controller fulfills its task. The control parameters specified in the controller DB must be adapted to the scan time.

You should always call OB251 from the time OB (OB13). You can set time OBs at a call up interval ranging between 10 ms and 655350 ms. The PID algorithm requires no more than 1 ms to process.

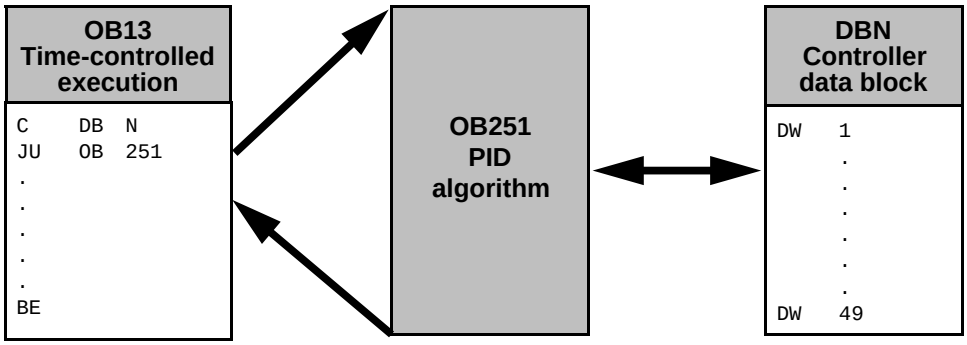


Figure 9-1. Calling Up the OB251 PID Algorithm

The continuous-action controller is designed for controlled systems such as those present in process engineering for controlling pressure, temperature, or flow rate.

The ”R” variable sets the proportional component of the PID controller.
If proportional action is required, most controller designs use the value R = 1.

The individual Proportional action, Integral action, and Derivative action components can be deactivated via their parameters (R, TI, and TD) by presetting the pertinent data words to zero. This enables you to implement all required controller structures without difficulty, e.g., PI, PD, or PID controllers.

You can forward the system deviation XW or, using the XZ input, any disturbance variable or the inverted actual value X to the derivative action element. Specify a negative K value for a reverse acting controller.

When the correction information (dY or Y) is at a limit, the integral action component is automatically deactivated in order not to impair the dynamic response of the controller.

The switch settings in the block diagram are implemented by setting the respective bits in control word ”STEU”.

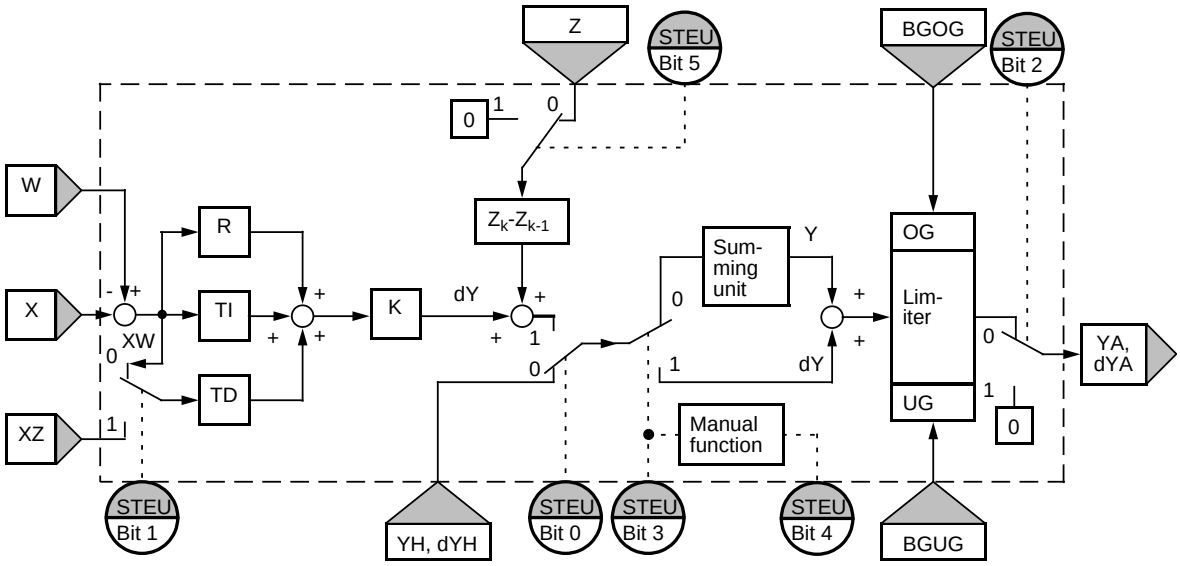


Figure 9-2. Block Diagram of the PID Controller

Table 9-3. Legend for the Block Diagram of the PID Controller (Figure 9-2)

Designation	Explanation		
K	Proportional coefficient:	K>0 K<0	direct acting reverse acting
R	R parameter (usually 1000)		
TA	Scan time		
TN	Integral-action time		
TV	Derivative-action time		
TI	Constant TI	TI=Scan time TA/Integral-action time TN	
TD	Constant TD	TD=Derivative-action time TV/Scan time TA	
W	Setpoint		
STEU	Control word		
YH, dYH	Output value:	YH dYH	Control Word Bit 3=0 Control Word Bit 3=1
Z	Disturbance variable		
XW	System error		
X	Actual value		
XZ	Substitute value for system error		
Y, dY	Manipulated variable, manipulated increments		
BGOG	Upper limit of the manipulated variable		
BGUG	Lower limit of the manipulated variable		
YA, dYA	Output value :	YA dYA	Control Word Bit 3=1 Control Word Bit 3=0

Table 9-4. Description of the Control Bits in Control Word "STEU"

Control Bit	Name	Signal State	Description
0	AUTO	0	Manual mode The following variables are updated in Manual mode: 1) X_K, XW_{K-1} and PW_{K-1} 2) XZ_K, XZ_{K-1} and PZ_{K-1}, when STEU bit 1=1 3) Z_K and Z_{K-1}, when STEU bit 5=0 Variable dD_{K-1} is set to 0: The algorithm is not computed.
		1	Automatic mode
1	XZ EIN	0	XW_k is forwarded to the differentiator. The XZ input is ignored.
		1	A variable other than XW_k is forwarded to the differentiator via the XZ input.
2	REG AUS	0 1	Normal controller processing When the controller is invoked (OB251), all variables (DW 18 to DW 48) with the exception of K, R, TI, TD, BGOG, BGUG, YH_k and W_k are reset in the controller DB. The controller is deactivated.
3	GESCHW	0 1	Positioning algorithm Correction rate algorithm
4	HANDART	0	When GESCHW=0: Following the transfer to Manual mode, the specified manipulated variable value YA is adjusted exponentially to the manual value in four sampling steps. Additional manual values are then forwarded immediately to the controller output. When GESCHW=1: The manual values are forwarded immediately to the controller output. The limiting values are in force in Manual mode.
		1	When GESCHW=0: The manipulated variable last output is retained. When GESCHW=1: Correction increment dY_K is set to zero.
5	NO Z	0 1	With feedforward control No feedforward control
6 and 7	-		These bits are not assigned.
8 to 15	-		The PID algorithm uses these bits as auxiliary flags.

The control program can be supplied with fixed values or parameters. Parameters are input via the assigned data words. The controller is based on a PID algorithm. Its output signal can be either a manipulated variable (positioning algorithm) or a manipulated variable modification (correction rate algorithm).

Correction Rate Algorithm

The relevant correction increment dY_k is computed at instant t= k · TA according to the following formula:

- Without feedforward control (D11.5=1); XW is forwarded to the differentiator (D11.1=0)

$dY_k = K [(XW_k - XW_{k-1}) R+TI \cdot XW_k + (TD (XW_k - 2XW_{k-1}+XW_{k-2})+dD_{k-1})]$

$= K (dPW_k R+dI_k+dD_k)$

- With feedforward control (D11.5=0); XW is forwarded to the differentiator (D11.1=0)

$dY_k = K [(XW_k - XW_{k-1}) R+TI \cdot XW_k + (TD (XW_k - 2XW_{k-1}+XW_{k-2})+dD_{k-1})]+(Z_k-Z_{k-1})$

$= K (dPW_k R+dI_k+dD_k)+dZ_k$

- Without feedforward control (D11.5=1); XZ is forwarded to the differentiator (D11.1=1)

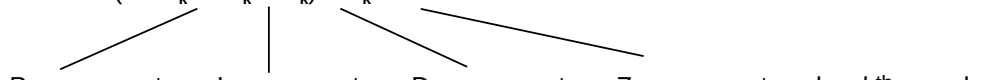
$dY_k = K [(XW_k - XW_{k-1}) R+TI \cdot XW_k + (TD (XZ_k - 2XZ_{k-1}+XZ_{k-2})+dD_{k-1})]$

$= K (dPW_k R+dI_k+dD_k)$

- With feedforward control (D11.5=0); XZ is forwarded to the differentiator (D11.1=1)

$dY_k = K [(XW_k - XW_{k-1}) R+TI \cdot XW_k + (TD (XZ_k - 2XZ_{k-1}+XZ_{k-2})+dD_{k-1})]+(Z_k-Z_{k-1})$

$= K (dPW_k R+dI_k+dD_k)+dZ_k$



P component I component D component Z component k: k^{th} sample

When XW_k is applied:	XW_k	=	$W_k - X_k$
	PW_k	=	$XW_k - XW_{k-1}$
	QW_k	=	$PW_k - PW_{k-1}$
		=	$XW_k-2XW_{k-1}+XW_{k-2}$
When XZ is applied:	PZ_k	=	$XZ_k - XZ_{k-1}$
	QZ_k	=	$PZ_k - PZ_{k-1}$
		=	$XZ_k-2XZ_{k-1}+XZ_{k-2}$
The result is:	dPW_k	=	$(XW_k- XW_{k-1})R$
	dI_k	=	$TI \cdot XW_k$
	dD_k	=	$(TD \cdot QW_k+dD_{k-1})$ when XW is applied
		=	$(TD \cdot QZ_k+dD_{k-1})$ when XZ is applied
	dZ_k	=	$Z_k - Z_{k-1}$

Positioning Algorithm

The formula used to compute the correction rate algorithm is also used to compute the positioning algorithm.

In contrast to the correction rate algorithm, however, the sum of all correction increments computed (in DW 48), rather than the correction increment dY_k is output at sampling instant t_k.

At instant t_k, manipulated variable Y_k is computed as follows:

$$Yk = \sum_{m=0}^{m=k} dYm$$

Initializing the PID Algorithm

OB251's interface to its environment is the controller DB.
All data needed to compute the next manipulated variable value is stored in this DB. Each controller must have its own controller data block.

The controller-specific data are initialized in a data block that must comprise at least 49 data words. The CPU goes to STOP with a transfer error (TRAF) if no DB has been opened or if the DB is too short.

> **Caution**
>
> Make sure that the right controller DB has been opened before calling control algorithm OB251.

Table 9-5. Format of the Controller DB

Data Word	Name	Comments
1	K	Proportional gain (-32 768 to + 32 767) for controllers without a differential comp. Proportional gain (- 1500 to +1500) for controllers with a diff. comp.[1] K is greater than zero when the control is direct acting, and less than zero when the control is reverse acting; the specified value is multiplied by a factor of 0.001.
3	R	R parameter (- 32 768 to+32 767) for controllers w/o a diff. comp. R parameter (- 1500 to+1500) for controllers with a diff. comp.[1] Normally 1 for controllers with P component; the specified value is multiplied by the approximate factor of 0.001
5	TI	Constant TI (0 to 9999) $TI = \frac{\text{Sampling interval TA}}{\text{Integral-action time TN}}$ The specified value is multiplied by the factor of 0.001.
7	TD	Constant TD (0 to 999) $TD = \frac{\text{Derivative-action time TV}}{\text{Sampling interval TA}}$
9	W	Setpoint (- 2047 to+2047)

1 It is possible to have larger gains, if sudden incremental changes to the system deviation are small enough. This is the reason you have to divide larger deviations into smaller ones such as adding the setpoint via a ramp function.

Table 9-5. Format of the Controller DB (Continued)

Data Word	Name	Comments
11	STEU	Control word (bit pattern)
12	YH	Value for manual operation (- 2047 to +2047)
14	BGOG	Upper limit value (- 2047 to +2047)
16	BGUG	Lower limit value (- 2047 to +2047)
22	X	Actual value (- 2047 to +2047)
24	Z	Disturbance variable (- 2047 to +2047)
29	XZ	Derivative time (- 2047 to +2047)
48	YA	Output variable (- 2047 to +2047)

All parameters (with the exception of the control word STEU) must be specified as 16-bit fixed point numbers.



Caution

The PID algorithm uses the data words that are not listed in Table 9-5 as auxiliary flags.

Initialization and Call Up of the PID Controller in a STEP 5 Program

Several different PID controllers can be implemented by calling up OB251 repeatedly. A data block must be initialized prior to each OB251 call up.

Note

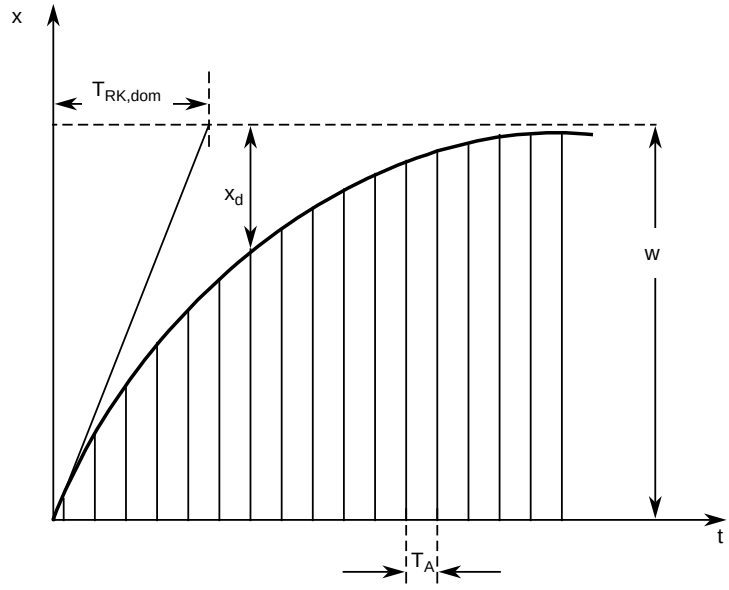
Important controller data are stored in the high-order byte of control word DW11 (DL11). Therefore make sure that only T DR 11/SU D11.0 to D11.7 or RU D 11.0 to D11.7 operations are used to modify user-specific bits in the control word.

Selecting the Sampling Interval

In order to be able to use the known analog method of consideration for digital control loops too, do not select a sampling interval that is too large.
Experience has shown that a TA sampling interval of approximately 1/10 of the time constant $T_{RK,\ dom}$* produces a control result comparable to the equivalent analog result. Dominant system time constant $T_{RK,\ dom}$ determines the step response of the closed control loop.

$$TA = 1/10 \cdot T_{RK,\ dom}$$

In order to ensure the constancy of the sampling interval, OB251 must always be called up in the service routine for time interrupts (OB13).



x = Control variable
t = Time
T_A = Sampling interval
$T_{RK,dom}$ = Dominant system time constant of the closed control loop
w = System error / Setpoint
x_d = Control deviation

Figure 9-3. Principle of Interval Sampling

* $T_{RK,\ dom}$=dominant system time constant of the closed control loop

Example for the Use of the PID Controller Algorithm:

A PID controller is supposed to keep an annealing furnace at a constant temperature.
The temperature setpoint is entered via a potentiometer.

The setpoints and actual values are acquired via analog channels 0 (IW 40) and 1 (IW 42) and forwarded to the controller. The computed manipulated variable is then output via the analog output channel (QW10).

The controller mode is set in input byte 32 (see control word DW 11 in the controller DB).
You must use the well-known controller design procedure to determine how to tune the controller for each controlled system.

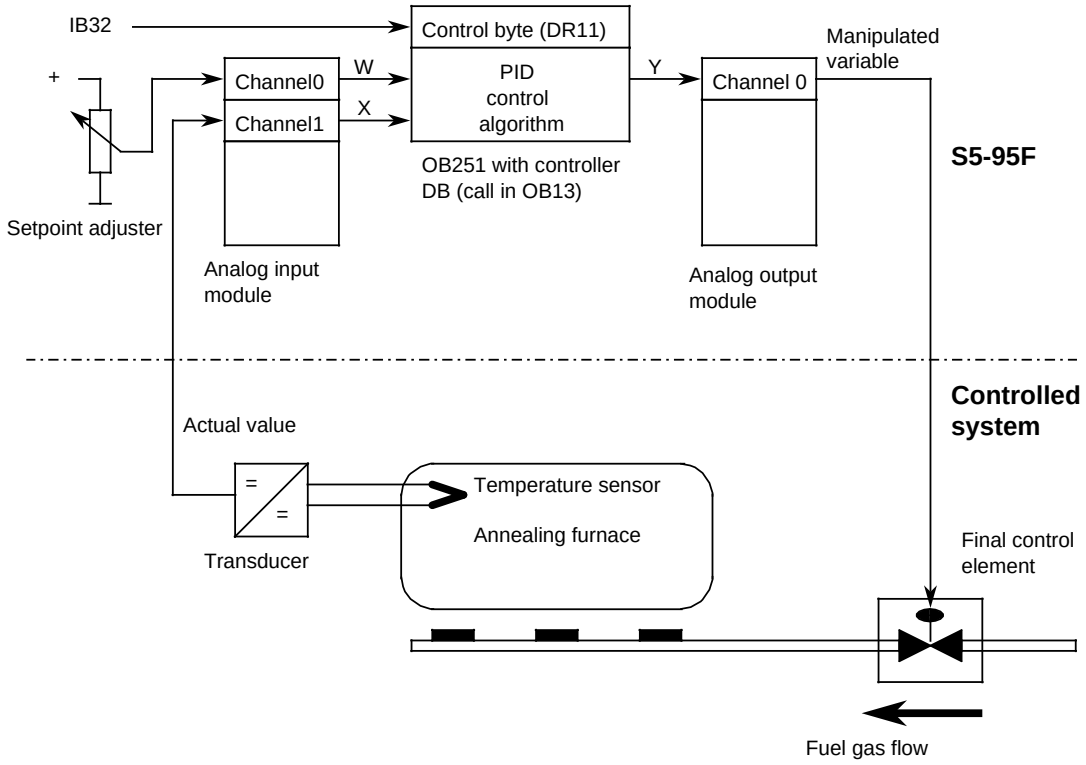


Figure 9-4. Process Schematic

The analog signals of the setpoint and actual values are converted into corresponding digital values in each sampling interval (set in OB13). OB251 uses these values to compute the new digital manipulated variable, from which, in turn, the analog output module generates a corresponding analog signal. This signal is then forwarded to the controlled system.

Calling the controller in the program:

OB13	STL	Description
	:	
	:JU FB 10	Process controller
NAME	:CONTROLLER 1	
	:	The controller's sampling interval
	:	depends on the time base used to call
	:	OB13 (set in DB1). When selecting the
	:	sampling interval, take into account the
	:BE	encoding time of the onboard analog inputs.

FB10	STL	Description
NAME	:CONTROLLER 1	
	:	
	:C DB 30	Select controller's DB
	:	
	:	**********
	:	Read controller's control bits
	:	**********
	:	
	:L PY 32	Read controller's control bits and store in
	:T MB 10	DR11
	:T DR 11	Note carefully:
	:	DR11 contains important control data for
	:	OB251,the control bits must therefore be
	:	transferred with T DR11 to prevent
	:	corrupting DL11
	:	
	:	**********
	:	Read actual value and setpoint
	:	**********
	:	
	:A F 12.0	Flag 0 (for unused functions in FB250)
	:R F 12.0	
	:AN F 12.1	Flag 1
	:S F 12.1	
	:	
	:JU FB 250	Read actual value
NAME	:RLG: AI	
BG	: KF 0	Slot number
KNKT	: KY 0,6	Channel no. 0, fixed-point bipolar
OGR	: KF +2047	Upper limit for actual value
UGR	: KF - 2047	Lower limit for actual value
EINZ	: F 12.0	No selective sampling
XA	: DW 22	Store scaled actual value in controller DB
FB	: F 12.2	Error bit
BU	: F 12.3	Range violation
	:	

FB10 (Continued) STL				Explanation
	:			
	:JU	FB	250	Read setpoint
NAME	:RLG:	AI		
BG	:	KF	+8	Slot number
KNKT	:	KY	1,6	Channel no. 1, fixed-point bipolar
OGR	:	KF	+2047	Upper limit for setpoint
UGR	:	KF	- 2047	Lower limit for setpoint
EINZ	:	F	12.0	No selective sampling
XA	:	DW	9	Store scaled setpoint in controller DB
FB	:	F	13.1	Error bit
BU	:	F	13.2	Range violation
	:			
	:A	F	10.0	In manual mode, the setpoint is set to the actual
	:JC	=WEIT		value to force the controller to
	:L	DW	22	react to a system deviation,
	:T	DW	9	if any, with a P step on transfer
	:			to Automatic mode.
	:			
WEIT	:			
	:			**********
	:	JU OB 251		Call controller
	:			**********
	:			
	:			**********
	:			Output value of manipulated variable Y
	:			**********
	:JU FB 251			
NAME	:RLG:AQ			
XE	:	DW	48	
BG	:	KF	1	Slot number
KNKT	:	KY	0,1	Channel 0, fixed-point bipolar
OGR	:	KF	+2047	Upper limit for actuating signal
UGR	:	KF	- 2047	Lower limit for actuating signal
FEH	:	F	13.5	Error bit when limiting value defined
BU	:	F	13.6	Manipulated variable Y to analog output
	:BE			Range violation

DB30	STL	Explanation
0:	KH = 0000;	
1:	KF = +01000;	K parameter (here=1), factor 0.001
2:	KH = 0000;	(value range: - 32768 to 32767)
3:	KF = +01000;	R parameter (here=1), factor 0.001
4:	KH = 0000;	(value range: - 32768 to 32767)
5:	KF = +00010;	TI=TA/TN (here=0.01), factor 0.001
6:	KH = 0000;	(value range: 0 to 9999)
7:	KF = +00010;	TD=TV/TA (here=10), factor 1
8:	KH = 0000;	(value range: 0 to 999)
9:	KF = +00000;	Setpoint w, factor 1
10:	KH = 0000;	(value range: - 2047 to 2047)
11:	KM = 00000000 00100000;	Control word
12:	KF = +00500;	Manual value YH, factor 1
13:	KH = 0000;	(value range: - 2047 to 2047)
14:	KF = +02000;	Upper controller limit BGOG, factor 1
15:	KH = 0000;	(value range: - 2047 to 2047)
16:	KF = -02000;	Lower controller limit BGUG, factor 1
17:	KH = 0000;	(value range: - 2047 to 2047)
18:	KH = 0000;	
19:	KH = 0000;	
20:	KH = 0000;	
21:	KH = 0000;	
22:	KF = +00000;	Actual value X, factor 1
23:	KH = 0000;	(value range: - 2047 to 2047)
24:	KF = +00000;	Disturbance variable Z, factor 1
25:	KH = 0000;	(value range: - 2047 to 2047)
26:	KH = 0000;	
27:	KH = 0000;	
28:	KH = 0000;	
29:	KF = +00000;	Feedforward XZ for difference,
30:	KH = 0000;	factor 1, (- 2047 to 2047)
31:	KH = 0000;	
32:	KH = 0000;	
33:	KH = 0000;	
34:	KH = 0000;	
35:	KH = 0000;	
36:	KH = 0000;	
37:	KH = 0000;	
38:	KH = 0000;	
39:	KH = 0000;	
40:	KH = 0000;	
41:	KH = 0000;	
42:	KH = 0000;	
43:	KH = 0000;	
44:	KH = 0000;	
45:	KH = 0000;	
46:	KH = 0000;	
47:	KH = 0000;	
48:	KF = +00000;	Controller output Y, factor 1
49:	KH = 0000;	(value range: - 2047 to 2047)
50:		

9.2 Integrated Function Blocks

The S5-95F has integral function blocks which makes use of the programmable controller more user-friendly. Function blocks FB230 to FB234, FB236 to FB238, FB240 to FB243 and FB252 to FB255 are failsafe. FB235, FB250 and FB251 are reaction-free. These blocks can be called in the control program with the commands ”JU FB x” or ”JC FB x”, where x is the block number.

Overview of Integrated Function Blocks

Block No.	Name	Function	Execution Time	See Section
FB230	HF:KOPPL	Data link for an H/F system	See section 16.2.6	16.2
FB231	HF:TBAS	Time base for H/F systems	17.5 ms	16.10
FB232	RLG:AE2	Reading in of two analog values with discrepancy analysis	5 ms	11.7.1
FB233	RLG:AE3	Reading in of three analog values with discrepancy analysis	7.5 ms	11.7.3
FB234	DE:2V3	2-out-of-3 evaluation for digital input modules	3 ms	9.2.1
FB235	TD/OP	Data link to operator panels and text displays	3 ms	9.2.2
FB236	DE:AV/LU	Non-equivalence and watchdog timers for digital input modules	1.5 ms	9.2.3
FB237	RLG:POLY	Linearization of a characteristic in a polygon	3.5 ms	9.2.4
FB238	HF:TIMER	Timer for H/F system	1 ms	16.10
FB240	COD:B4	Code converter for changing BCD into fixed-point	0.5 ms	9.2.5
FB241	COD:16	Code converter for changing fixed-point into BCD	0.6 ms	9.2.6
FB242	MUL:16	Multiplication of two fixed-point numbers	0.6 ms	9.2.7
FB243	DIV:16	Division of two fixed-point numbers	1.4 ms	9.2.8
FB250	RLG:AE	Reading in of analog values	1.6 ms	11.6.1
FB251	RLG:AA	Output of analog values	3.3 ms	11.6.2
FB252	AGF:TEST	Processing of test components	See Table 9-32	9.2.9
FB255	AGF:DEPA	Depassivation of I/Os	5 to 20 ms	9.2.10

Exceptions for Function Blocks FB 230 to FB 238

> **Note**
>
> Integrated function blocks FB 230 to FB 238 use flags in the range from FY 200 to FY 250 (scratchpad). If you use these function blocks, you may use flags FY 200.0 to FY 255.7 in your user program only if you save this flag area before calling one of these function blocks and reload it after the function block has executed.

9.2.1 2-out-of-3 Evaluation for Failsafe Digital Inputs -FB234-

This block increases the availability of binary input signals. 2-out-of-3 evaluation allows the process to continue when one of the three input signals fails. Depending on the chosen configuration, this feature can increase only the availability of the input modules or also of the sensors.

The block can be used for quality levels 4, 5 and 6. It is possible to implement configurations with one, two or three sensors (see configuration overview).

Block Parameters for FB234

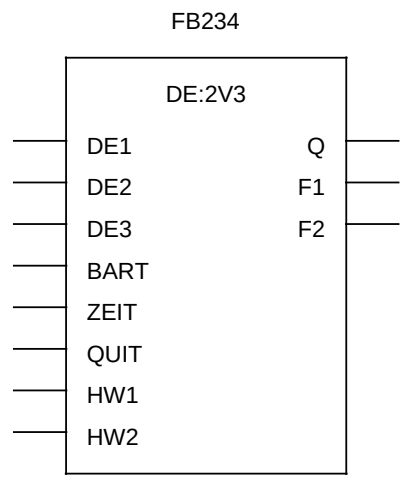
STL	CSF
0000 :JU FB 234 0001 NAME :DE:2V3 0002 DE1 : 0003 DE2 : 0004 DE3 : 0005 BART : 0006 ZEIT : 0007 QUIT : 0008 HW1 : 0009 HW2 : 000A Q : 000B F1 : 000C F2 : 000D :BE	

Table 9-6. Block Parameters for FB234

Parameter	Description	Type		Contents	Remarks
DE1	Digital input 1	I	BI	I0.0 to I33.7	*2)
DE2	Digital input 2	I	BI	I0.0 to I33.7	*2)
DE3	Digital input 3	I	BI	I0.0 to I33.7	*2)
BART	Block mode as relates to error evaluation and error acknowledgement	D	KF	0 = Without system error evaluation 1 = With system error evaluation 2 = With system acknowledgement during operation	
ZEIT	Specification for discrepancy time in number of OB1 cycles	D	KF	0 to +32767	
QUIT	Acknowledgement for error outputs F1, F2	I	BI	I0.0 to I33.7, F0.0 to F199.7	
HW1	Internal auxiliary word for time counting	I	W	Free, retentive flag word FW0 to FW62	*1)
HW2	Internal auxiliary word for storing block status info	I	W	Free, retentive flag word FW0 to FW62	*1)
Q	Output bit for unified digital inputs	Q	BI	Binary output or flag Q0.0 to Q34.3, F0.0 to F199.7	
F1	Output bit for primary error (acceptance status 1)	Q	BI	Binary output or flag Q0.0 to Q34.3, F0.0 to F199.7	
F2	Output bit for secondary error (acceptance status 1)	Q	BI	Binary output or flag Q0.0 to Q34.3, F0.0 to F199.7	

*1) Only retentive flag words may be used as internal auxiliary locations, the reason being so that internal error status and current discrepancy times will not be inadvertantly reset in the event of an unforeseen power outage. These flag words must be preset to 0 in the cold restart branch (OB21). These flag words must not be reset in the warm restart branch in the event of a power outage (OB22).

*2) In order that it be possible for the two remaining signals to be read in via different subunits should one of the three signals fail, it is necessary that at least one of the inputs be a two-channel failsafe input. Of the other two non-failsafe inputs, one must be in subunit A and one in subunit B. It is also possible for all inputs to be two-channel failsafe inputs. Should it be necessary to work with flags or outputs at input parameters DE1 to DE3 for special applications, it is, in fact, possible. In this case, scanning of the relevant input parameter for system errors is automatically deactivated, regardless of the mode.

Operands Reserved by FB234

Table 9-7. Reserved Operands and Blocks

Type	Reserved Area
Temporary flags	FY240 to FY255
Internally called blocks	DB254 (system event DB)

> **Safety Note**
>
> The programmed discrepancy time (parameter time)must always be less than or equal to the process error tolerance time.

FB234 Cycle Time Requirements

FB234's cycle time requirement depends on the selected mode.

Table 9-8. Cycle Time Requirements

Mode	Cycle Time Requirements
0	Typically < 2 ms
1	Typically < 3 ms
2	Without acknowledgement cycle: Typically < 3 ms With acknowledgement cycle: Max. 22 ms (depending on the number of acknowledged error messages)

FB234 Functionality

The block performs a discrepancy analysis for the three binary input signals DE1 to DE3 and generates a uniform output signal Q.
The output signal corresponds to the status of the input signals as long as the input signals have the same status.

When one of the input signals changes, the discrepancy timer is started. During this time, the last valid value is output to output Q. If the signal states differ when the discrepancy time runs out, the status of the two identical inputs is forwarded to output Q and a primary error report via output F1 (output F1 goes to 0). The primary error is stored. The input in question is no longer evaluated.

When another error occurs in addition to the first, a secondary error is flagged at the function block's F2 output (output F2 goes to 0).
At the same time, the block's Q output is set to 0.

Depending on the mode setting, the block can sometimes also evaluate errors flagged by the operating system. This evaluation, however, is active for inputs only:
When flags or outputs are initialized at the block's signal inputs, errors flagged by the operating system are not evaluated for the signal inputs in question.

Safety Note

Output F1 should normally generate an interrupt.
Evaluation of output F2 or of output Q if set to 0 for this error situation, must result in a shutdown of the affected part of the system or to a safe state.

Modes

Function block parameter BART is used to select one of three different modes.

Table 9-9. Modes

BART Parameter	Description
Mode 0	The parameter monitors the states of the input signals for equality. In the event of an error, the faulted input is phased out when the discrepancy time is over. As a result, the relevant signals F1 (primary error) and F2.Q (secondary error) are reset at the block output. Errors flagged by the operating system are not evaluated in this mode. You can use this mode for monitoring of the inputs and an appropriate error response (such as STOP or passivation) by the operating system. When an error occurs and when an error has been eliminated, the block can be returned to the acceptance state via the acknowledgement input.
Mode 1	The basic function is the same as in mode 0. In addition, the block monitors the errors flagged by the operating system regarding the inputs. Monitoring always extends to the entire input byte in which a parameterized input is located. When an error is detected, all inputs in that byte are immediately marked as errored, and are phased out of the processing sequence. As a result, the relevant signals F1 (primary error) and F2.Q (secondary error) are reset at the block input. In this mode, it is possible to initialize the error response for failsafe inputs as user response. The user response is handled by the function block in this case. This mode thus increases the availability of the entire system. In this mode, pressing the acknowledgement button does not reset system errors. You must organize the resetting of the bits in the static error image of DB254 and the clearing of the error messages in the message buffer of DB254 yourself by means of FB255.
Mode 2 (for signal groups with AND or OR logic only)	The basic function is the same as in mode 1. Once errors have been eliminated and the acknowledgement button pressed, however, all system flags pertaining to the inputs initialized in the block are reset. This increases the availability of the function configured with the block. Resetting of the system flags during operation takes a noticeable amount of cycle time (see dynamic block response). Use this mode when this additional cycle time load is justifiable for your control program. For reasons of safety, the block resets only those errors relating to the inputs initialized in the block. Should other inputs be marked as errored in the input bytes in which the initialized inputs are located, the block does not reset any of their error states. You must organize this yourself. You must also see to resetting passivated signal groups and the DIAG LED on the basic units yourself with the aid of FB255. Note that FB255 must be called before FB234.

Safety Note

If you are working in mode 1, you must reset the static error bits in DB254 yourself. It is absolutely necessary that, along with the resetting of the error bits in the static I/O error image, the associated error blocks in DB254's error stack also be reset.
This ensures that the operating system can flag a new error on the same input module.

Additional Monitoring of System Errors in Modes 1 and 2

In modes 1 and 2, function block FB234 also responds selectively to specific error flags from the operating system.

For two-channel failsafe or single-channel inputs, the block responds to the following operating system flags.

Table 9-10. Errors Reported by the Operating System

Error Message	Two-Channel Failsafe Inputs
Hardware fault on onboard DI	Error 40
Hardware fault on external DI module	Error 44
Discrepancy time over for onboard DI	Error 50
Discrepancy time over for external DI	Error 54
Short-circuit on sensor line to onboard DI	Error 57
Short-circuit on sensor line to external DI	Error 59
Error Message	**Single-Channel Inputs**
Hardware fault on external DI module	Error 44

The module test for single-channel input modules is active only when failsafe input modules of type 6ES5 431-8FAxx are used here as well.

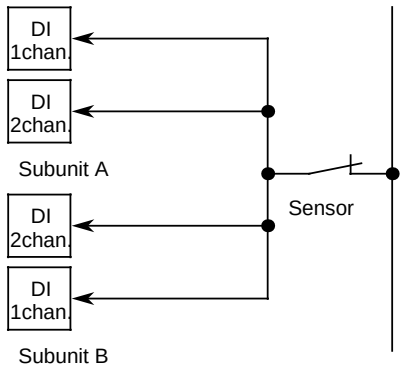
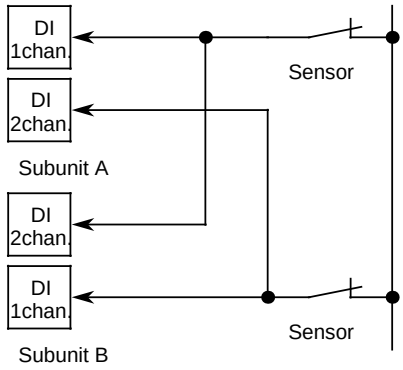
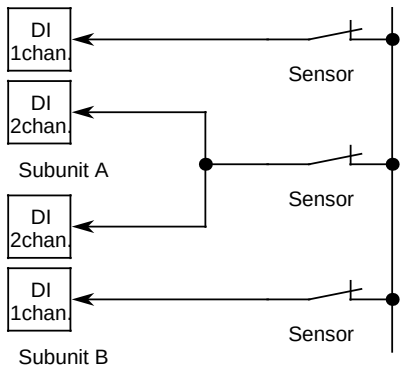
To evaluate the tests executed by the operating system, the error bits for the relevant input module are analyzed in diagnostic block DB254.

In the event of an error reported by the operating system, the relevant input is no longer evaluated and F1 (for a primary error) or F2 (for a secondary error) immediately generated.

For reasons of safety, the entire input byte in which an input is located is evaluated for every input initialized in the block.
If an input is errored, the entire input byte is regarded as errored.

Overview of Possible Configurations for the 2-out-of-3 Evaluation with FB234

Table 9-11. Overview of Possible Configurations for the 2-out-of-3 Evaluation with FB234

Wiring Diagram	Remarks
I/O type H1 	**Quality level to DIN V 19250** Quality level 4, 5, 6 **Sensor type** Failsafe sensors **Error response to sensor failure** Sensor failure cannot be detected. **Error response to failure of a DI** Primary errors are reported. The installation can continue to operate.
I/O type H2 	**Quality level to DIN V 19250** Quality level 4, 5, 6 **Sensor type** Non-failsafe sensors **Error response to sensor failure** Primary and secondary errors are reported. The installation must be shut down. **Error response to failure of a DI** Primary errors are reported. The installation can continue to operate.
I/O type H3 	**Quality level to DIN V 19250** Quality level 4, 5, 6 **Sensor type** Non-failsafe sensor **Error response to sensor failure** Primary errors are reported. The installation can continue to operate. **Error response to failure of a DI** Primary errors are reported. The installation can continue to operate.

Safety Note

- 6ES5 431-8FAxx failsafe modules must be used in the configurations discussed above. This also applies to single-channel inputs.
- The configurations discussed above satisfy the specified quality levels with the prerequisite that the input signals be intermittent, that is to say, the signals must assume each state at least once within the secondary error occurrence time for the duration of the specified discrepancy time.
  If the process signals do not fulfill this condition, you must use two-channel failsafe DIs for each of FB234's three inputs, whereby the sensors must be supplied via short-circuit test outputs.
  The functionality of DB234 and 2-out-of-3 evaluation remain unchanged.

Parameter Initialization Example for FB234

STL	Description
OB1	
NETWORK 1 0000	
0000 :JU FB 234	2 of 3 digital inputs
0001 NAME :DE:2V3	
0002 DE1 :I 0.0	Failsafe DI subunit A/B
0003 DE2 :I 2.0	Single-channel DI subunit A
0004 DE3 :I 3.0	Single-channel DI subunit B
0005 BART :KF +2	Mode 2 with system error evaluation
0006 ZEIT :KF +5	Discrepancy time 5 OB1 cycles
0007 QUIT :F 8.0	Flag for acknowledgement pulse
0008 HW1 :FW 4	Retentive auxiliary flag word 1
0009 HW2 :FW 6	Retentive auxiliary flag word 2
000A Q :F 10.0	Standardized output signal
000B F1 :F 10.1	Flag primary error
000C F2 :F 10.2	Flag secondary error
000D :BE	

When necessary, retentive auxiliary flag words HW1 and HW2 can be reset in the manual cold restart branch (OB21).
It is, however, not permissible to reset these auxiliary flag words in the automatic cold restart branch (OB22) because the stored error states of the block must be retained in the event of a voltage drop.

9.2.2 Interfacing Operator Panels and Text Displays via the CP 521 SI's Serial Port -FB235-

Function block FB235 supports data interchange between the S5-95F and a text display or operator panel that is plugged into a CP 521 SI's serial port. The data exchange is handled in free ASCII protocol (FAP).

The block supports the following TD/OP functions:

- Output of process values (output fields)
- Output of status messages
- Output of fault messages
- PG functions: STAT VAR for data blocks
- PG functions: CONTROL VAR for input DB
- Image of the function keyboard in input DB
- Image of system keyboard in input DB
- Entry of non-failsafe setpoint values in the input DB (input fields, input/output fields)

The block is reaction-free. The data entered via a TD or OP will be transferred only to the specified input DB (I-DB parameter).
Any post-processing of the data entered is the responsibility of the programmer.

All data entered via the OP must be checked for valid and non-critical values before being processed in the user program. The same rules apply here as for the use of the parameter control DB (see section 18.16).

In contrast to the parameter control DB, however, data can also be entered while the process is active. The required checking of safety-related input data must therefore be in the cyclic part of the program. Data not within the permissible range may not be forwarded to the user program.

FB235 Block Parameters

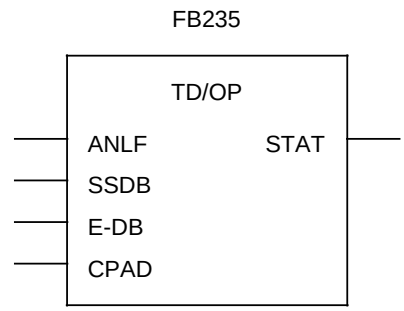
STL	CSF
0000 :JU FB 235 0001 NAME :TD/OP 0002 ANLF : 0003 SSDB : 0004 E-DB : 0005 CPAD : 0006 STAT : 0007 :BE	

Table 9-12. FB235 Block Parameters

Parameter	Description	Type		Contents	Remarks
ANLF	Start bit	I	BI	F 0.0 ... F 199.7	1)
SSDB	Number of the interface data block	D	KF	2 to 250 (not DB51)	2)
E-DB	Number of the input data block	D	KF	2 to 250 (not DB51)	3)
CPAD	CP start address	D	KF	64, 72, 80, 88, 96, 104, 112, 120	4)
STAT	Status word	Q	W	FW 0 to FW 198	5)

1) The start bit must be set in the start OBs. It is reset by FB235.

2) The interface data block must be created by the user in a length of at least 170 data words (DW 0 to DW 169). It contains the internal workarea for FB235 as well as Send and Receive buffers. The user program has no Write access to the interface data block.
The block must be preset to KH 0000.

3) The input data block must be created in the required length. Its length depends on the number of configured input fields (see the manual for the relevant configuring tool, such as ProTool or COM TEXT).
When no input DB is needed (no input fields), a 0 must be entered here. Input data block and interface data block must have different DB numbers.

4) The CP 521 SI reserves eight contiguous bytes in the programmable controller's input/output area.

5) The status word contains four status flags and an error number.

Safety Note

- The reaction-free characteristics of FB235 can be ensured only when the block parameters are correctly initialized.
- The CP 521 SI, which is required to drive the OP, may not be initialized/used to output system messages to a printer.

Bits in Status Word STAT

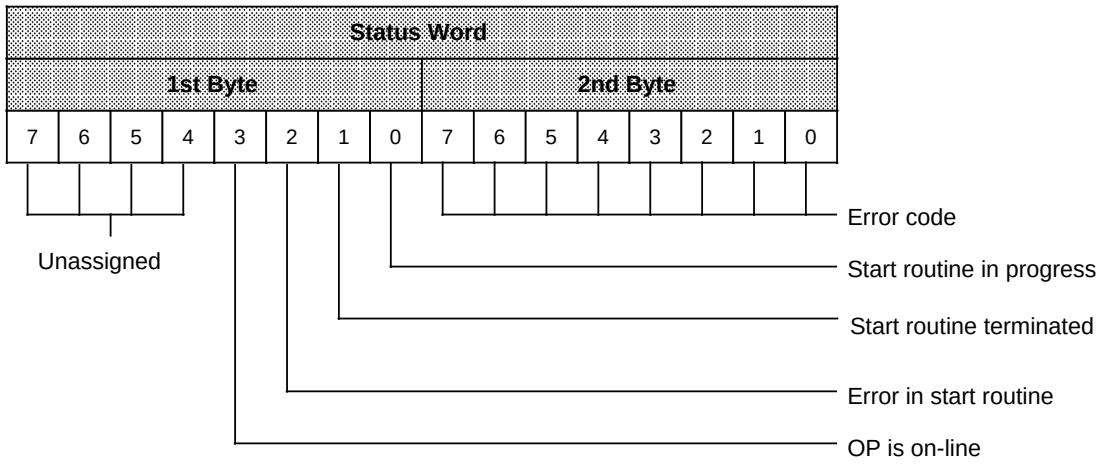


Figure 9-5. The Status Word

The ”Start routine in progress” bit is set when the parameter check in the start routine has been completed. It is reset when the CP 521 SI has been initialized.

The ”Start routine terminated without error” bit is set when the CP 521 SI has been initialized.

In order to monitor the link between OP and S5-95F, the TD/OP inverts a ”ready bit” in each frame. FB235 counts the number of times it is called between this and the next inversion of the ”ready bit”. The ”on-line” bit is set when the ready bit was inverted. It is reset when it was not inverted within the next 300 calls.

The block deposits one of the following error codes in the start routine:

Table 9-13. Error Codes in the Status Word

Error Code	Description
00	No errors.
01	The DB number of the interface DB is lower than 2.
02	The DB number of the interface DB is higher than 250.
03	No interface DB.
04	The interface DB is too short.
10	CP start address cannot be divided by 8.
11	CP start address is lower than 64.
12	CP start address is higher than 120.
20	DB number of the input DB is lower than 2.
21	DB number of the input DB is higher than 250.
22	Interface DB and input DB have the same DB number.

If an error was detected in the start routine (error code not zero), no further blocks are processed. The error must be rectified in the user program and the start bit re-set.

Execution Time and Reserved Operands

The execution time of FB235 is approximately 3 ms. FB235 reserves the operands listed below.

Table 9-14. FB235 Reserved Operands

Type	Reserved Area
Temporary flags	FY200 to FY255

Configuring Information

In order to ensure problem-free communication between the CP 521 SI and the OP, FB 235 must execute approximately every 100 ms.
If the program cycle time is greater than 100 ms, you should therefore invoke FB235, with the same parameters, several times in the cycle.

In order for the data to be forwarded to the CP 521 SI between these calls, you must also program the following command sequence im OB13 (timed interrupt).

STL	Description
OB 13 NETWORK 1 0000 0000 :L PW CPAD 0001 :T IW CPAD 0002 :L PW CPAD+2 0003 :T IW CPAD+2 0004 :L PW CPAD+4 0005 :T IW CPAD+4 0006 :L PW CPAD+6 0007 :T IW CPAD+6 0008 : 0009 :L QW CPAD 000A :T PW CPAD 000B :L QW CPAD+2 000C :T PW CPAD+2 000D :L QW CPAD+4 000E :T PW CPAD+4 000F :L QW CPAD+6 0010 :T PW CPAD+6 0011 :BE	

CPAD is a spaceholder for the CP 521 SI's base address programmed in FB235.

OB13 can then be configured so that it is called at regular intervals, for example every 80 ms.

Alternatively to being called in OB1, FB235 can be called in OB13.
When FB235 is called in OB13, it may not be called in the cyclic program.

Output of Process Values, Status Messages and Fault Messages

General functional sequence: The initiative for data interchange always comes from the OP. The OP sends data request frames to the S5-95F for the output fields, status messages and fault messages configured with ProTool/COM TEXT at regular intervals (polling time).
If the data are available, the S5-95F sends them in a positive acknowledgement frame to the OP, where they are output to a display or updated.
If the data are not available, a negative acknowledgement with error message, e.g. ”No DB”, is sent to the OP. In this case, a system error message is output to the display.

When configuring output fields, status messages and fault messages with COM TEXT, note that FB235 accepts as data source only data blocks with DB numbers between 2 and 250 (excluding DB51). All other data requests (such as FWs, IWs, QWs) receive a negative acknowledgement. Each time data are requested, FB235 checks to see if there is such a data block and if so, whether it fulfills the length requirements. The smallest possible amount of data that can be sent is a data word.

Display update times:

A maximum of six bytes of useful data can be transferred over the I/O interface between CP 521 SI and S5-95F per cycle. This fact produces the following guidelines for the update times for output fields:

Updating of a display with one output field:

Request frame from OP15 to S5-95F:	4 PLC cycles
Acknowledgement frame with data from S5-95F to OP 15:	4 PLC cycles
Total:	8 PLC cycles

Updating of a display with five output fields:

Request frame from OP15 to S5-95F:	8 PLC cycles
Acknowledgement frame with data from S5-95F to OP 15:	5 PLC cycles
Total:	13 PLC cycles

Exceptions when Configuring the Text Display/Operator Panel

Note

While configuring with ProTool/COM TEXT, you must specify a data block (DB TD/OP) to be used for the interchange of internal data (such as version numbers and device identifiers). For safety reasons, however, the S5-95F does not support this function. But because the configuring software requires an entry, you must enter DB51. The S5-95F flags an error if you enter any other block because FB235 checks the parameters for validity.
Because DB51 is not needed for internal data interchange, it is not necessary to make it available in the S5-95F.

Entering Data via the OP

When an OP is used to enter a setpoint value in a configured input field, the OP sends a data distribution frame to the S5-95F. FB235 accepts as destination only the input DB. Data distribution frames to other destinations are rejected (negative acknowledgement). The smallest unit of data which can be transferred is a data word.

Safety Note

- All data entered via the OP must be checked for validity and non-critical values prior to processing in the user program.
- The same rules apply here as for the use of parameter control DBs. In contrast to a parameter control DB, however, data can also be entered via the OP while the process is active. The required checking of safety-related input data must therefore be done, in this case, in the cyclic program section.
  Data outside the permissible range may not be passed to the control program.
- FB235 has write access to the entire input DB. No program data other than input data may be entered in this data block.
- Should the function key or system key image be evaluated in the user program, it (or they) may be used to control only non-critical process functions.

Error Blocks

Function block FB235 also supports the display of the error messages which the operating system enters in DB254. Each error message entered consists of an error block comprising eight data words (see section 15.3.8).

You may, if you wish, specify an error block number between 1 and 16 in DW 0 of the input DB. Error block number 1 corresponds to the newest, error block number 16 to the oldest error entry made by the S5-95F's operating system in DB254. If you want to specify an error block number, you can do so e.g. via an input field on the OP.

Error location, system response and primary and secondary information about the selected error block are stored in data words DW 160 to 165 of the interface DB. You can display the information on the OP in output fields.

Note

Errors which caused the S5-95F to go to STOP cannot be displayed because FB235 has to be processed in the program cycle in order to output error block information.

Table 9-15. The Interface DB

Data Word	Description
DW 160	Error location / System response D 160.0 - D 160.1 - D 160.2 User response as per DB1 parameter initialization (AR) D 160.3 Message (M) D 160.4 - D 160.5 - D 160.6 Error in subunit A (A) D 160.7 Error in subunit B (B) D 160.8 .. 160.15 -
DW 161	Primary information
DW 162	Secondary information
DW 163	Secondary information
DW 164	Secondary information

Error location and system response can be displayed in an output field of type ”symbolic actual value”. The following field texts, for instance, can be defined with ProTool/COM TEXT.

Table 9-16. Example for Defining Field Texts

Value	Field Text	Description
000		
068	B AR	Subunit B, user response
072	B M	Subunit B, message
132	A AR	Subunit A, user response
136	A M	Subunit A, message
196	AB AR	Subunit A and B, user response
200	AR M	Subunit A and B, message

The primary info can also be displayed on the OP as symbolic actual value, for example in two consecutive output fields:

Table 9-17. Output of S5-95F Error Messages via Field Texts

Value	Field Text 1 Field Text 2
000	00:No entry :
005	05:HW failure of integ. clock :
013	13:Error in user program :
040	40:HW fault on o nboard DI :
041	41:HW fault on o nboard DQ :
042	42:HW fault, onb . HW interrupt :
043	43:HW fault, onb oard counter :
044	44:HW fault, ext ernal DI module:
045	45:HW fault, ext ernal DQ module:
048	48:DB1 incorr. c onfigured :
050	50:Disc.time ove r f.onboard DI :
052	52:Disc.time ove r f.onboard DI :
053	53:Disp.time ove r f.onb.counter:
054	54:Disc.time ove r f.external DI:
057	57:Short-cir. on sensor line :
058	58:Short-cir. on sensor line :
059	59:Short-cir.on sensor line :
060	60:SINEC L1 time out :
061	61:SINEC-L1 ok :
062	62:SINEC-L1 fram e error :
063	63:SINEC-L1 send el.invalid :
064	64:SINEC-L1 data path depass. :
073	73:Battery empty , shutd. in 72h:

Example: Configuring the OP15 for Output of Error Blocks

The ProTool file (FB235D.O15) in the COM 95F software package contains a configuring example for the output of error blocks on the OP15's display. The display can be selected using the OP15's K1 function key. The error block numbers range between 1 (newest error block) and 16 (oldest error block). Using the cursor and the Enter key, the user can page through the error blocks.

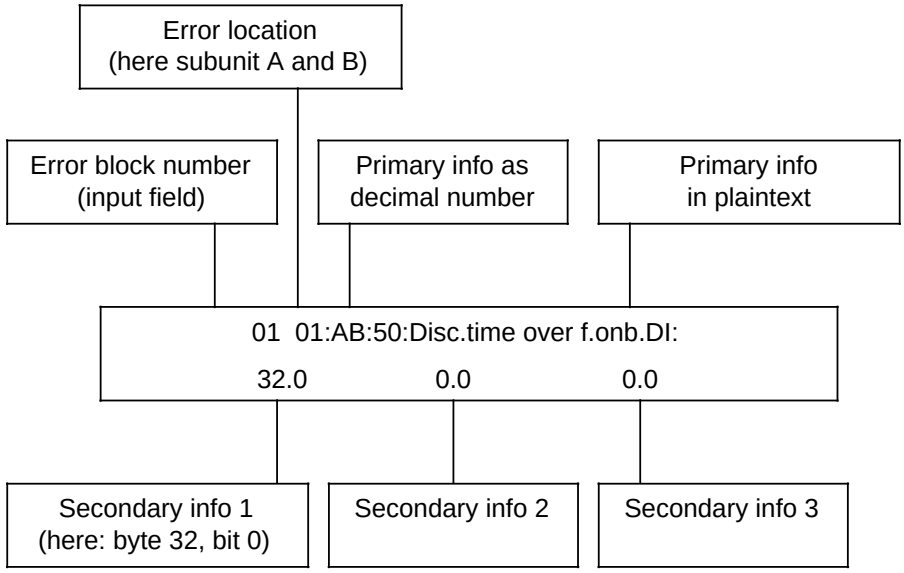


Figure 9-6. Example for Output of an Error Block on the OP15

If this file is used without modification, DB100 must be specified as input data block E-DB when initializing FB235 because DW 0 in DB100 was configured as process link for the error block number.

The OP15 also contains an image of the system and function keyboard.
The process link is defined as follows:

System keyboard image:	DB100, DW 1 and 2
Function keyboard image:	DB100, DW 3 and 4

Assignment or configuration can be changed as required.

Settings for CP 521 SI and OP

FB 235 requires the following fixed parameter values as transfer parameters for the CP 521 SI:

Baud rate:	9600
Parity:	Even
Busy signal:	No
Interface:	TTY
Data format:	1 start bit, 8 data bits, 1 parity bit, 1 stop bit
HW handshake:	No
Character delay:	10 ms
Maximum frame length:	88 bytes

Care must be taken that these parameters are correctly set when initializing the OP. The preset character delay (120 ms) for the OP must be retained. FAP (free ASCII protocol) must be specified as transmission protocol.

Parameter Initialization Example for FB 235

STL	Description
OB 21/OB 22	
NETWORK 1 0000	
0000 :O F 100.0	
0001 :ON F 100.0	
0002 :S F 100.0	Set start bit
0003 :BE	
OB 1	
NETWORK 1 0000	
0000 :JU FB 235	Link to OP
0001 NAME :TD/OP	
0002 ANLF :F 100.0	Start bit
0003 SSDB :KF +235	Interface DB
0004 E-DB :KF +100	Input data block
0005 CPAD :KF +64	CP start address
0006 STAT :FW 102	Status word
0007 :BE	

Note

Note that, in addition to the sample parameters shown above, OB13 must also be programmed. Data interchange with the CP 521 SI must be organized in OB13 (see the information presented in ”Configuring Information”).

9.2.3 Function Block FB236 for Non-Equivalence and Watchdog Timers

FB236 monitors two digital inputs for non-equivalence or for equivalence. The block can be used for evaluation of non-equivalent sensor signals or as watchdog timer for actuators.

The block is approved for quality levels 4, 5 and 6. Depending on what is needed, two-channel failsafe inputs or single-channel inputs may be used for the input signals (see Tables 9.20 and 9.21).

FB236 Block Parameters

STL

```
0000         :JU FB     236
0001  NAME   :DE:AV/LU
0002  E1     :
0003  E2     :
0004  A/L    :
0005  QUIT   :
0006  ZEIT   :
0007  Q1     :
0008  DISK   :
0009  HW1    :
000A  HW2    :
000B         :
000C         :BE
```

CSF

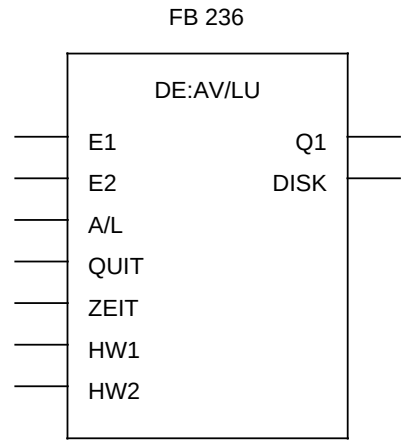


Table 9-18. FB236 Block Parameters

Parameter	Description	Type		Contents	Remarks
E1	Binary input 1	I	BI	I, Q, F	
E2	Binary input 2	I	BI	I, Q, F	
A/L	FB mode selection	D	KC	A for non-equivalence L for watchdog timer	The DISK signal is reset if setting not A or L
QUIT	Error messages	I	BI	I, Q, F	
ZEIT	Discrepancy time in OB1 cycles	D	KF	KF=0 to 32767	
Q1	Binary output allocated to binary input E1	Q	BI	I, Q, F	When non-equivalence ”1”, when E1=1 and E2=0, when watchdog ”1”, when E1 and E2 = 0
DISK	Discrepancy error	Q	BI	I, Q, F	Acceptable state ”1”. If there is a discrepancy error and the discrepancy time is over, the latching output is set to ”0”
HW1	Internal auxiliary flag with edge evaluation and error storage	Q	W	FW 0 to FW62 or DW	*1)
HW2	Auxiliary word for time counting	Q	W	FW 0 to FW62 or DW	*1)

*1) If a flag word is used, it must be a retentive flag word. If a data word is used, the data block must be opened before FB236 is called.

Execution Time and Operands Reserved by FB236

FB236’s execution time is approximately 1.5 ms. The FB reserves the following operands:

Table 9-19. Reserved Operands

Type	Reserved Area
Temporary flags	FY 201 to FY 207

Non-Equivalence Test/Watchdog Mode

Although these two functions are used at different locations in the program, the internal handling of the input signals is almost the same.

The A/L parameter is used to choose between non-equivalence mode (A) or watchdog mode (L). If a letter other than A or L is specified, FB236 reports a fault and resets output Q1 and the discrepancy indicator DISK.

If there is a discrepancy between input signals E1 and E2, a watchdog timer (discrepancy time in OB1 cycles) expires. The discrepancy timer is started as soon as one of the input signals changes its value. When the timer expires, the output and the discrepancy indicator are reset. While the discrepancy timer is running, the last valid value is forwarded to the output. The value for the discrepancy time is specified by the ZEIT parameter, and is given in OB1 cycles.

A non-equivalence error is detected when the states of the two input signals are equal after the discrepancy time has expired.

A watchdog detected when the states of the two input signals are not equal after the discrepancy time has expired.

Non-equivalence Test

When it performs a non-equivalence test, FB236 generates a "1" signal at output Q1 when E1 = "1" and E2 = "0". Vice versa, when E1 = "0" and E2 = "1", output Q1 is set to "0".

If there is no malfunction, the "DISK" output is "1". If there is a malfunction (inputs E1 and E2 are both "1" or both "0"), output Q1 and the "DISK" output are reset when the discrepancy time has run out.

This state can be acknowledged via the QUIT input when the input signals are once again correct.

States of the inputs and outputs when the discrepancy time has expired.

E2	E2	Q1	DISK	Result
0 1	1 0	0 1	1 1	Acceptance state
0 1	0 1	0 0	0 0	Error

Watchdog Function

In the case of the watchdog function, FB236 generates a "1" signal at output Q1 when E1 = "1" and E2 = "1". In the reverse case, output Q1 is set to "0" when E1 = "0" and E2 = "0".

If there is no malfunction, the "DISK" output is set to "1". If there is a malfunction (different signals at inputs E1 and E2), output Q1 and the "DISK" output are reset when the discrepancy time has run out.

This state can be acknowledged via the QUIT input when the input signals are once again correct.

States of the inputs and outputs when the discrepancy time has expired.

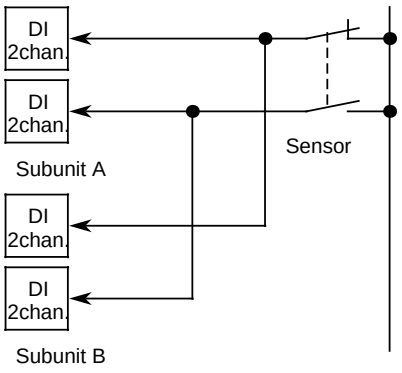
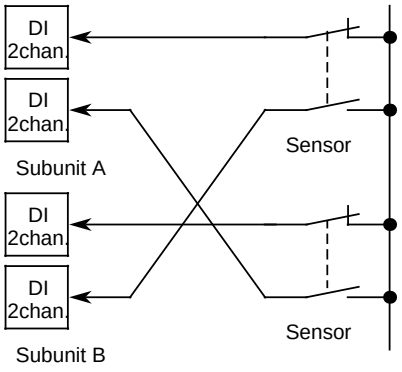
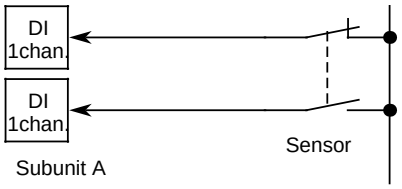
E2	E2	Q1	DISK	Result
1 0	1 0	1 0	1 1	Acceptance state
1 0	0 1	0 0	0 0	Error

Safety Note

The block itself does not evaluate module faults/errors.
The two-channel failsafe inputs must be monitored either by the operating system (Stop) or by programming an appropriate user response.

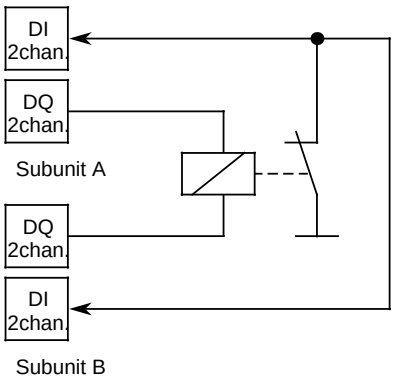
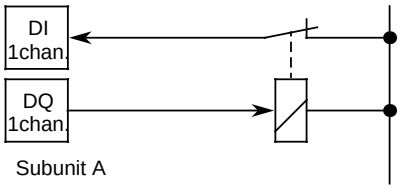
Configurations for Non-Equivalence Tests

Table 9-20. Overview of Possible Configurations for FB236 (Non-Equivalence Tests)

Schematic Diagram	Description
I/O type G.1 	**Quality level to DIN V 19250** Quality level 4, 5, 6 **Sensor type** Non-equivalence evaluation with a non-equivalent sensor via failsafe two-channel inputs.
I/O type G.2 	**Quality level to DIN V 19250** Quality level 4, 5, 6 **Sensor type** Non-equivalence evaluation with two non-equivalent sensors via failsafe two-channel inputs.
I/O type G.3 	**Quality level to DIN V 19250** Quality level: None **Sensor type** Simple non-equivalence evaluation with a non-equivalent sensor or non-failsafe single-channel inputs.

Configurations for Watchdog Timers

Table 9-21. Overview of Possible Configurations for FB236 (Watchdog Timers)

Schematic Diagram	Description
I/O type G.4  	**Quality level to DIN V 19250** Quality level 4, 5, 6 **Sensor type** Watchdog with two-channel failsafe outputs and two-channel failsafe inputs.
I/O type G.5  	**Quality level to DIN V 19250** Quality level: None **Sensor type** Simple watchdog timer with non-failsafe single-channel output and non-failsafe single-channel input.

Parameter Initialization Example for a Non-Equivalence Evaluation

STL	Description
OB 1 E:BSP236ST.S5D	LAE=18
NETWORK 1 0000	
0000 :JU FB 236	
0001 NAME :DE:AV/LU	
0002 E1 : I 2.0	Binary input 1
0003 E2 : I 2.1	Binary input 2
0004 A/L : KS A	Non-equivalence testing mode
0005 QUIT : I 1.0	Input for malfunction acknowledgements
0006 ZEIT : KF +5	Discrepancy time is 5 OB1 cycles
0007 Q1 F 9.0	Block output
0008 DISK : F 9.1	Discrepancy indicator
0009 HW1 : FW 10	Auxiliary word 1
000A HW2 : FW 14	Auxiliary word 2
000B :	
000C :BE	

9.2.4 Function Block FB237, Interpolation Block

This function block converts normalized input values between -2048 and +2048 into non-linear output values which may also lie in the range from -2048 to +2048. Such a conversion may be necessary when the input value is made available by a sensor with a non-linear characteristic (such as a PT100 resistance thermometer). One or two data blocks are needed to define the interpolation points of the non-linear function.
The data blocks may be used more than once for different input signals with the same characteristic.

FB237 Block Parameters

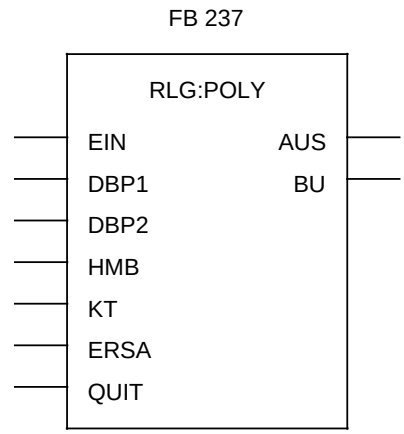
STL	CSF
0000 :JU FB 237 0001 NAME :RLG:POLY 0002 EIN : 0003 QUIT : 0004 HMB : 0005 DBP1 : 0006 DBP2 : 0007 KT : 0008 ERSA : 0009 AUS : 000A BU : 000B :BE	

Table 9-22. FB237 Block Parameters

Parameter	Description	Type		Contents	Remarks
EIN	Input value	I	W	FW0 to FW198 Value range: -2048 to +2048	
DBP1	Data block containing the interpolation points for the positive input values	B		DB2 to DB250	
DBP2	Data block containing the interpolation points for the negative input values	B		DB2 to DB250	
HMB	Auxiliary flag byte for storing internal states	I	W	FY0 to FY199	
KT	Unipolar or bipolar representation	D	KF	0 = Unipolar (EIN = 0 to 2048) 1 = Bipolar (EIN =-2048 to 2048)	
ERSA	Substitute value for range violations	D	KF	Value range: KF=-32768 to 32767	
QUIT	Acknowledgement input for errors with internal edge evaluation	Q	BI	I0.0 to I32.7 F0.0 to F199.7	
AUS	Output value	Q	W	FW 0 to FW 198 Value range: -2048 to +2048	
BU	Range violation	Q	BI	F0..0 to F199.7 Q0.0 to Q34.3 1= Acceptance state	

Additional Remarks About Specific Parameters

Table 9-23. Additional Remarks Regarding Block Parameters

Parameter	Description
DBP1,DBP2	The data blocks must be declared as CONSTANT in COM 95F. Each data block must comprise at least 23 data words for representation of a rough characteristic. If you want to represent precise characteristic sections, you need an additional 17 data words plus 10 data words for each precise section. The data block for negative input values is not needed if representation is to be unipolar, in which case DB0 must be specified as spaceholder for the DPB2 parameter. If several analog inputs with the same characteristic are to be evaluated, the same interpolation points can be programmed in all FB237 calls.
BU	A range violation is triggered in the following cases: • In bipolar mode when the input value is not between -2048 and 2048. • In unipolar mode when the input value is not between 0 and 2048. • When the input value exceeds the limits defined in DBP1/DBP2. • When a value other than 0 or 1 was specified for the KT parameter. • When an output value outside the range -2048 to 2048 would result because of incorrect initialization of the interpolation point DBs. The range violation is latching. When there is once again a valid input value, the range violation can be reset by a positive edge at the acknowledgement input. When a range violation is triggered, the programmed substitute value is always forwarded to output AUS.

Execution Time and Operands Reserved by FB237

FB237's execution time is approximately 3.5 ms. FB237 reserves the following operands:

Table 9-24. Reserved FB237 Operands

Type	Reserved Area
Temporary flags	FY 200 to FY 239

FB237 Functionality

This block calculates the function values of a non-linear function y = f (x).
The interpolation points of the non-linear function are in data blocks DBP1 (for positive x values) and DBP2 (for negative x values).
The block executes a linear interpolation for values which lie between the interpolation points.

If you initialize the block for unipolar mode (positive x values only), you need only data block DBP1.

In bipolar mode (positive and negative x values), you need data block DBP1 only.

The block can process a rough characteristic consisting of up to 17 interpolation points in the positive and negative range.
The interpolation points for the positive range are allocated to the x values (0, 128, 256, ..., 2048), the interpolation points in the negative range corresponding to the x values (0, -128, -256, ..., -2048).

In addition, you may define one or more precise sections for the rough characteristic.

A precise section must always lie between two adjacent interpolation points of the rough characteristic.
To make this possible, you must define seven interpolation points for each precise characteristic. The first and last interpolation point of the precise characteristic correspond to the assigned interpolation points of the rough characteristic.
The areas of a precise characteristic section each have a constant range of 16 units.

You can specify an upper limit and a lower limit for the input value in data blocks DBP1 and DBP2.
This allows you to restrict the permissible range of the input signal.
When an input value exceeds/falls short of the specified limit, the BU (range violation) signal is reset and the substitute value specified as ERSA parameter is output.

The lower limit/upper limit has no effect on the allocation or grid of the interpolation points for the rough characteristic.
The rough characteristic must be fully defined over the entire range between the upper and lower limits.
If your lower limit or upper limit does not coincide with an interpolation point, you must also define the interpolation point located outside the characteristic.

If you want to work with precise characteristic sections, you must define a list of starting points for the individual precise characteristic sections beginning DW 22 in DBP1/DBP2. The x value of an interpolation point in the rough characteristic is always starting point for a precise characteristic section.
Permissible values for DBP1 are thus: 0, 128, 256, ..., 1920.
Permissible values for DBP2 are: 0, -128, -256, ..., -1920.
The starting point is always the point of the precise characteristic section closest to the coordinate origin.

The list has a variable length and always ends with end identifier KH FFFF.

If you do not want to use precise characteristic sections, you must enter this end identifier directly into DW 22.

The sequence of the entries in the list is arbitrary. Each list entry, however, is allocated to a data area with a length of 10 data words defining the interpolation points for the precise characteristic section.
For the first list entry, this data area begins at DW 40, for the second list entry at DW 50, and so on.

Safety Note

Function block FB237 does not carry plausibility checks of the interpolation point parameters.
It is up to you to see to it that the interpolation point blocks are properly initialized, and that defined output values can be determined for all possible input values.
Precise characteristic sections always require all seven interpolation points.
For the acceptance inspection, the output values must be checked for each interpolation point.

Interpolation Point Data Blocks DBP1/DBP2

Table 9-25. Data Blocks DBP1/DBP2 for the Rough Characteristic

Data Word	Contents	Description
DW 0	KH=0000	Unassigned
DW 1	KH=0000	Unassigned
DW 2	KF=Xu	Lower limit of the input value for the rough characteristic (e.g. 0 / -2048)
DW 3	KF=Xo	Upper limit of the input value for the rough characteristic (e.g. 2048 / 0)
DW 4	KF=Y0	Rough char. interpolation point for x0 = 0
DW 5	KF=Y1	Rough char. interpolation point for x1 = 128 / -128
DW 6	KF=Y2	Rough char. interpolation point for x2 = 256 / -256
DW 7	KF=Y3	Rough char. interpolation point for x3 = 384 / -384
DW 8	KF=Y4	Rough char. interpolation point for x4 = 512 / -512
DW 9	KF=Y5	Rough char. interpolation point for x5 = 640 / -640
DW 10	KF=Y6	Rough char. interpolation point for x6 = 768 / -768
DW 11	KF=Y7	Rough char. interpolation point for x7 = 896 / -896
DW 12	KF=Y8	Rough char. interpolation point for x8 = 1024 / -1024
DW 13	KF=Y9	Rough char. interpolation point for x9 = 1152 / -1152
DW 14	KF=Y10	Rough char. interpolation point for x10 = 1280 / -1280
DW 15	KF=Y11	Rough char. interpolation point for x11 = 1408 / -1408
DW 16	KF=Y12	Rough char. interpolation point for x12 = 1536 / -1536
DW 17	KF=Y13	Rough char. interpolation point for x13 = 1664 / -1664
DW 18	KF=Y14	Rough char. interpolation point for x14 = 1792 / -1792
DW 19	KF=Y15	Rough char. interpolation point for x15 = 1920 / -1920
DW 20	KF=Y16	Rough char. interpolation point for x16 = 2048 / -2048
DW 21	KH=0000	Unassigned
DW 22	KH=FFFF	End identifier (no PK list)

The values at the right of the slash are always those for DBP2

Interpolation Point Data Blocks DBP1/DBP2 for Precise Characteristic Sections

Table 9-26. Data Blocks DBP1/DBP2 for Precise Characteristic Sections

Data Word	Contents	Description
DW0...21		Same as Table 9-25
DW 22	KF=x0	Starting point x0 of the 1st precise characteristic section (PK1)
DW 24	KF=x0	Starting point x0 of the 2nd precise characteristic section (PK2)
...	...	...
DW xx	KF=FFFF	List end identifier
...	...	...
DW 39	KH=0000	Unassigned
DW 40	KF=y0	Interpolation point from PK1 for x = x0
DW 41	KF=y1	Interpolation point from PK1 for x = x0+16 / x0-16
DW 42	KF=y2	Interpolation point from PK1 for x = x0+32 / x0-32
DW 43	KF=y3	Interpolation point from PK1 for x = x0+48 / x0-48
DW 44	KF=y4	Interpolation point from PK1 for x = x0+64 / x0-64
DW 45	KF=y5	Interpolation point from PK1 for x = x0+80 / x0-80
DW 46	KF=y6	Interpolation point from PK1 for x = x0+96 / x0-96
DW 47	KF=y7	Interpolation point from PK1 for x = x0+112 / x0-112
DW 48	KF=y8	Interpolation point from PK1 for x = x0+128 / x0-128
DW 49	KH=0000	Unassigned
DW 50	KF=y0	Interpolation point from PK2 for x = x0
DW 51	KF=y1	Interpolation point from PK2 for x = x0+16 / x0-16
DW 52	KF=y2	Interpolation point from PK2 for x = x0+32 / x0-32
DW 53	KF=y3	Interpolation point from PK2 for x = x0+48 / x0-48
DW 54	KF=y4	Interpolation point from PK2 for x = x0+64 / x0-64
DW 55	KF=y5	Interpolation point from PK2 for x = x0+80 / x0-80
DW 56	KF=y6	Interpolation point from PK2 for x = x0+96 / x0-96
DW 57	KF=y7	Interpolation point from PK2 for x = x0+112 / x0-112
DW 58	KF=y8	Interpolation point from PK2 for x = x0+128 / x0-128
DW 59	KH=0000	
...	...	...Unassigned

The values at the right of the slash are always those for DBP2

Sample Parameters

The non-linear function illustrated below is to be simulated by a rough characteristic in the range x = -384 to 1792 and by a precise characteristic section im the range x = 1408 to 1536.
The precise characteristic section is a close-up of subsection x = 1409 to 1536 of the rough characteristic.

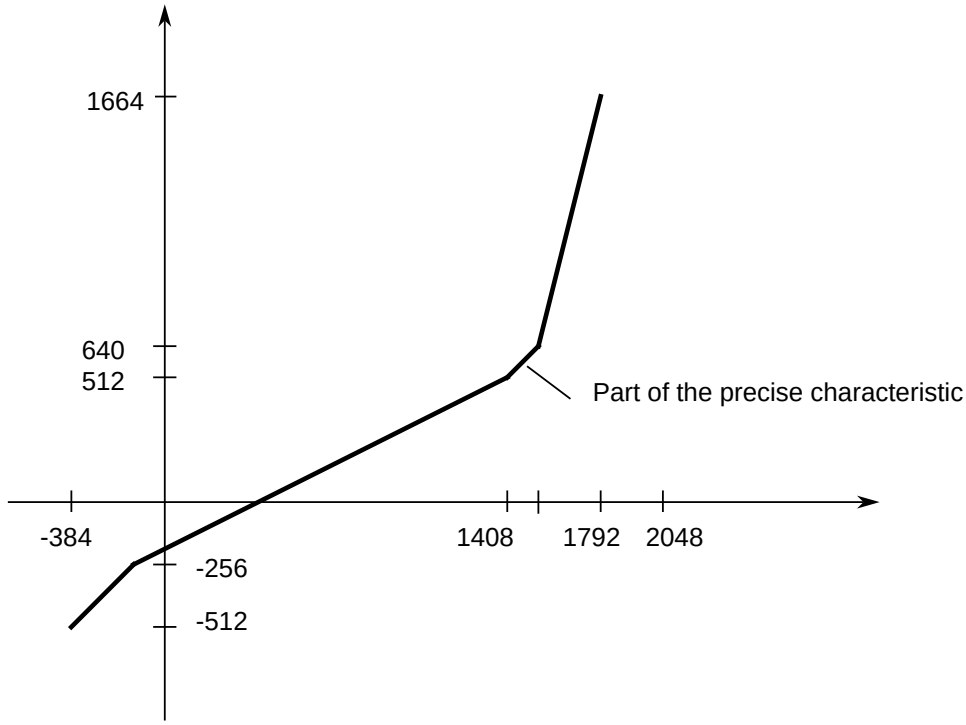


Figure 9-7. Function Characteristic

Value Table for the Characteristic

Rough Characteristic	
x	y
1792	1664
1664	1152
1536	640
1408	512
1280	448
1152	384
1024	320
896	256
768	192
640	128
512	64
384	0
256	-64
128	-128
0	-192
-128	-256
-256	-384
-384	-512

Precise Characteristic	
x	y
1536	640
1520	576
1504	560
1488	544
1472	528
1456	524
1440	520
1424	516
1408	512

Figure 9-8. Value Table

Because the input value can also assume negative values, you must use FB237's bipolar representation option. For this reason, you will need two data blocks to define the interpolation points.

Because the precise characteristic section is located in the positive range of x, this characteristic section is initialized in DBP1.

The starting point of the precise characteristic is x = 1408.
DBP2 contains no precise characteristic sections.

The lower limit for DBP1 is 0, the upper limit is 1792.
The lower limit for DBP2 is -384, the upper limit is 0.

User Program for the Sample Parameters

STL				Description
OB 1		E:ANWBSPST.S5D		LAE=17
NETWORK 1		0000		
0000		:JU FB 237		Call interpolation block
0001	NAME	:RLG:POLY		
0002	EIN	:	FW 10	Input signal
0003	QUIT	:	I 2.0	Acknowledgement input
0004	HMB	:	FY 20	Auxiliary flag byte
0005	DBP1	:	DB 237	Interpolation point DB for positive x
0006	DBP2	:	DB 238	Interpolation point DB for negative x
0007	KT	:	KF +1	Characteristic type is bipolar
0008	ERSA	:	KF -512	Substitute value
0009	AUS	:	FW 12	Output signal
000A	BU	:	F 21.0	Range violation
000B		:BE		

STL	Description
DB237 E:ANWBSPST.S5D	LAE=55 /18
0: KH = 0000; Unassigned	
1: KH = 0000; Unassigned	
2: KF = +00000;	Lower limit f. input value
3: KF = +01792;	Upper limit f. input value
4: KF = -00192;	Interpolation point for x=0
5: KF = -00128;	Interpolation point for x=128
6: KF = -00064;	Interpolation point for x=256
7: KF = +00000;	Interpolation point for x=384
8: KF = +00064;	Interpolation point for x=512
9: KF = +00128;	Interpolation point for x=640
10: KF = +00192;	Interpolation point for x=768
11: KF = +00256;	Interpolation point for x=896
12: KF = +00320;	Interpolation point for x=1024
13: KF = +00384;	Interpolation point for x=1152
14: KF = +00448;	Interpolation point for x=1280
15: KF = +00512;	Interpolation point for x=1408
16: KF = +00640;	Interpolation point for x=1536
17: KF = +01152;	Interpolation point for x=1664
18: KF = +01664;	Interpolation point for x=1792
19: KF = +00000;	Interpolation point unassigned
20: KF = +00000;	Interpolation point unassigned
21: KH = 0000;	Unassigned
22: KF = +01408;	Starting value for PK1
23: KH = FFFF;	End of PK list
24: KH = 0000;	Entry not assigned
25: KH = 0000;	Entry not assigned
26: KH = 0000;	Entry not assigned
27: KH = 0000;	Entry not assigned
28: KH = 0000;	Entry not assigned
29: KH = 0000;	Entry not assigned
30: KH = 0000;	Entry not assigned
31: KH = 0000;	Entry not assigned
32: KH = 0000;	Entry not assigned
33: KH = 0000;	Entry not assigned
34: KH = 0000;	Entry not assigned
35: KH = 0000;	Entry not assigned
36: KH = 0000;	Entry not assigned
37: KH = 0000;	Entry not assigned
38: KH = 0000;	Entry not assigned
39: KH = 0000;	Unassigned
40: KF = +00512;	Interpolation point PK1 for x=1408
41: KF = +00516;	Interpolation point PK1 for x=1424
42: KF = +00520;	Interpolation point PK1 for x=1440
43: KF = +00524;	Interpolation point PK1 for x=1456
44: KF = +00528;	Interpolation point PK1 for x=1472
45: KF = +00544;	Interpolation point PK1 for x=1488
46: KF = +00560;	Interpolation point PK1 for x=1504
47: KF = +00576;	Interpolation point PK1 for x=1520
48: KF = +00640;	Interpolation point PK1 for x=1536
49: KH = 0000;	Unassigned
50:	

STL	Description
DB238 E:ANWBSPST.S5D	LAE=28/10
0: KH = 0000;	Unassigned
1: KH = 0000;	Unassigned
2: KF = -00384;	Lower limit f. input value
3: KF = +00000;	Upper limit f. input value
4: KF = -00192;	Interpolation point for x=0
5: KF = -00256;	Interpolation point for x=-128
6: KF = -00384;	Interpolation point for x=-256
7: KF = -00512;	Interpolation point for x=-384
8: KF = +00000;	Interpolation point unassigned
9: KF = +00000;	Interpolation point unassigned
10: KF = +00000;	Interpolation point unassigned
11: KF = +00000;	Interpolation point unassigned
12: KF = +00000;	Interpolation point unassigned
13: KF = +00000;	Interpolation point unassigned
14: KF = +00000;	Interpolation point unassigned
15: KF = +00000;	Interpolation point unassigned
16: KF = +00000;	Interpolation point unassigned
17: KF = +00000;	Interpolation point unassigned
18: KF = +00000;	Interpolation point unassigned
19: KF = +00000;	Interpolation point unassigned
20: KF = +00000;	Interpolation point unassigned
21: KH = 0000;	Unassigned
22: KH = FFFF;	No PK list
23:	

9.2.5 Code Converter : B4 - FB240 -

Use function block FB240 to convert a number in BCD (4 tetrads) with sign to a fixed-point binary number (16 bits).
You must change a two-tetrad number to a four-tetrad number before you convert it.

- If a tetrad is not in the BCD defined range, then FB240 displays the value ”0”. An error bit message does not follow.

Table 9-27. Calling and Parameter Assignments of FB240

Parameter	Meaning	Type	Assignment	STL
BCD	BCD number	I W	0 ... 9999	: JU FB 240 NAME : COD:B4 BCD : SBCD : DUAL :
SBCD	Sign of the BCD number	I BI	”1” for ”-” ”0” for ”+”	
DUAL	Fixed-point number (KF)	Q W	16 bits ”0” or ”1”	

9.2.6 Code Converter : 16 - FB241-

Use function block FB 241 to convert a fixed-point binary number (16 bits) to a number in BCD code with additional consideration of the sign.
An eight-bit binary number must be transferred to a 16-bit word before conversion.

Table 9-28. Calling and Parameter Assignments of FB241

Parameter	Meaning	Type	Assignment	STL
DUAL	Binary number	I W	- 32768 to +32767	: JU FB 241 NAME : COD:16 DUAL : SBCD : BCD2 : BCD1 :
SBCD	Sign of the BCD number	Q BI	"1" for "-" "0" for "+"	
BCD2	BCD no. 4th and 5th tetrads	Q BY	2 tetrads	
BCD1	BCD number tetrads 0 to 3	Q W	4 tetrads	

9.2.7 Multiplier : 16 - FB242 -

Use function block FB 242 to multiply one fixed-point binary number (16 bits) by another.
The product is represented by a fixed-point binary number (32 bits).

The result is also scanned for zero. An eight-bit number must be transferred to a 16-bit word prior to multiplication.

Table 9-29. Calling and Parameter Assignments of FB242

Parameter	Meaning	Type	Assignment	STL
Z1	Multiplier	I W	- 32768 to+32767	: JU FB 242 NAME : MUL:16 Z1 : Z2 : Z3=0 : Z32 : Z31 :
Z2	Multiplicand	I W	- 32768 to+32767	
Z3=0	Scan for zero	Q BI	"1" if the product is zero	
Z32	Product high-word	Q W	16 bits	
Z31	Product low-word	Q W	16 bits	

9.2.8 Divider : 16 - FB243 -

Use function block FB 243 to divide one fixed-point binary number (16 bits) by another. The result (quotient and remainder) is represented by two fixed-point binary numbers (16 bits each).

The divisor and the result are also scanned for zero. An eight-bit number must be transferred to a 16-bit word prior to division.

Table 9-30. Calling and Parameter Assignments of FB243

Parameter	Meaning	Type	Assignment	STL
Z1	Dividend	I W	- 32768 to+32767	: JU FB 243 NAME : DIV:16 Z1 : Z2 : OV : FEH : Z3=0 : Z4=0 : Z3 : Z4 :
Z2	Divisor	I W	- 32768 to+32767	
OV	Overflow bit	Q BI	"1" if overflow	
FEH		Q BI	"1" for division by zero	
Z3=0	Scan for zero	Q BI	"1" : quotient is zero	
Z4=0	Scan for zero	Q BI	"1" : remainder is zero	
Z3	Quotient	Q W	16 bits	
Z4	Remainder	Q W	16 bits	

9.2.9 Additional Calling Up of Test Routines - FB252 -

The operating system tests all safety-relevant system components at least once per hour (see Chapter 2). In the following section, this test is also called background test. If your automated process requires shorter test intervals for particular system components or if you wish to determine the instant of a test yourself, then you use FB252 to execute additional tests. The test routines are executed immediately after the FB252 is called up. You call up FB252 only in the cyclic program (OB1).

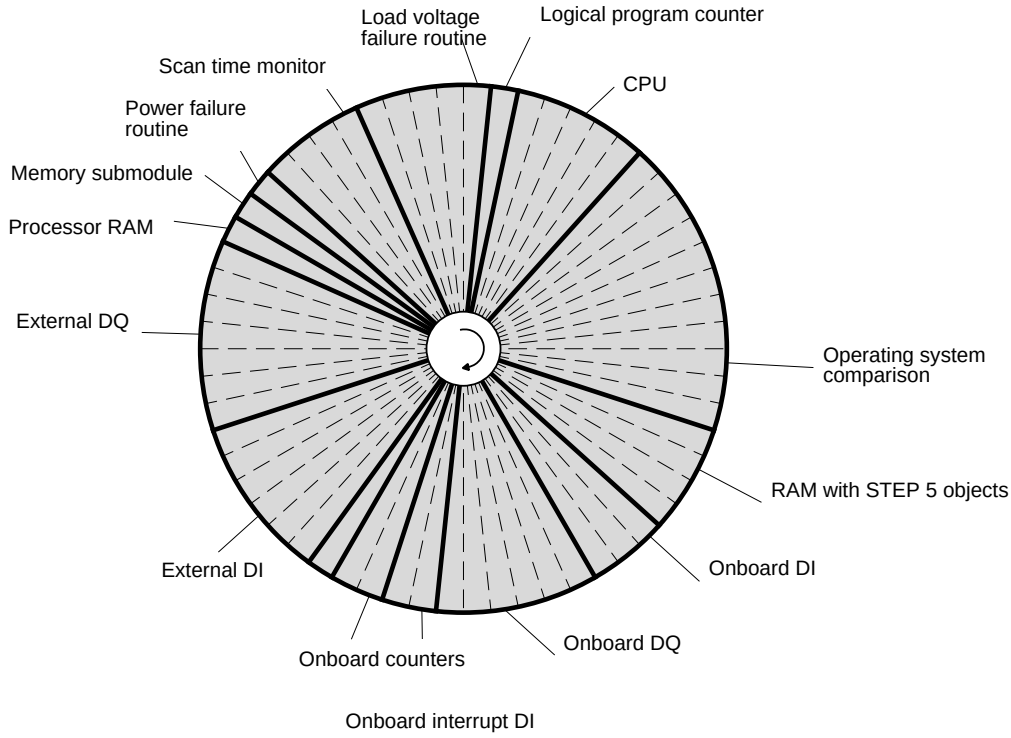


Figure 9-9. List of S5-95F Self-Test Components

Note

Because the testing of components lengthens the scan time, you should, if possible, invoke only the tests for the failsafe digital output modules (test numbers 04_H and 08_H) and short circuits (test number $0F_H$) in the user program.

Execution of the Background Test

All test components of the background test are stored in a list and are automatically called and executed by the operating system. Processing of the tests is organized so that all tests are executed once per hour.

Tests with runtimes of more than 5 ms (see Table 9-32) are subdivided by the S5-95F into several test segments, so that processing of one test component can be distributed to several PLC cycles. This distribution to several PLC cycles minimizes fluctuations of the PLC cycle times occurring as a result of the tests.

The load on the cycle time through tests is low. With a PLC cycle time of 200 ms, the S5-95F processes a test segment of approx. 5 ms in each cycle. The runtime of a test segment is approx. 30 ms in the case of complex I/O tests only.

> **Note**
>
> If you do not call up function block FB252, the S5-95F executes all test components automatically except for the short-circuit test (AUFT:$0F_H$).

Calling Up Test Components in the Control Program

If you use function block FB252 to explicitly request a test, then the S5-95F executes the requested test immediately. The background test initiated by the operating system is not affected by the test initiated by the user.

You may only activate one user-initiated test at a time. Before you activate a new test (e.g. in another part of the program), the last test must be completed. If not, none of the tests started will be completed. You should therefore always evaluate the information returned in the parameter assignment error byte.

Calling Up a Test Component for DQ Test only in the Control Program

Since the outputs are briefly disabled during the DQ test (see section 4.3), it is possible that quick-reacting actuators drop out unintentionally during operation. That is why you ***should not*** have the S5-95F execute the DQ test ***automatically*** (background test), but rather via function block FB252 when the process is in a defined, non-critical state.

You should therefore call FB252 at non-critical states with parameter BED = 1 (see Table 9-31). If these non-critical process states occur at intervals < 1 hour, the S5-95F will no longer execute the DQ test in the background. The first time the process reaches a non-critical state (within the one-hour test cycle), the DQ test is carried out and is not repeated in the current test cycle.

Depending on the parameterization of FB252, the DQ test is executed in one PLC cycle (parameter EINZ=0) or is distributed to several PLC cycles (parameter EINZ=1).

Table 9-31. Calling and Parameter Assignments of FB252

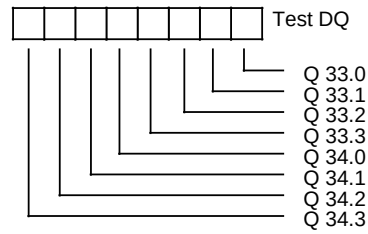
Parameter	Meaning	Type	Assignment	STL
AUFT	Number of test component	I/BY	00_H - CPU 01_H - Operating system comparison 02_H - RAM with STEP 5 objects 03_H - Onboard DI 04_H - Onboard DQ 05_H - Onboard interrupt 06_H - Onboard counter 07_H - External I/Os, DI 08_H - External I/Os, DQ 09_H - Processor RAM $0A_H$ - Memory submodule $0B_H$ - Power failure routine $0C_H$ - Scan time monitor (watchdog) $0D_H$ - Load voltage failure routine $0E_H$ - Logical program counter $0F_H$ - Short-circuit test	: JU FB 252 NAME : AGF:TEST AUFT : TYP : NR : BED : EINZ : PAFE :
TYP	Relevant only if AUFT = 02_H tested RAM area, in which the specified STEP 5 object lies	I/W	T = Timers C = Counters F = Flags PI = Process image tables OB = Organization blocks PB = Program blocks SB = Sequence blocks FB = Function blocks DB = Data blocks	
NR	Relevant only if AUFT = 02_H and block (OB to DB) specified with TYP or AUFT = $0F_H$	I/BY	For AUFT = 2 and TYP = B (block) 0 to 255 = Block number For AUFT = 0FH (short-circuit test) Bit mask for byte number  	

Table 9-31. Calling and Parameter Assignments of FB252 (continued)

Parameter	Meaning	Type	Assignment	STL
BED	Execution of test if it has not yet been executed in the current cycle	I/BI	0 = Execute test unconditionally 1 = Execute test if it has not yet been executed in the current test cycle	
EINZ	Distribute test to several PLC cycles	I/BI	0 = Execute complete test in one PLC cycle 1 = Distribute test to several PLC cycles	
PAFE	Message byte	I/BY	00_H = Test executed completely 41_H = Faulty FB252 parameter assignment $D0_H$ = Test executing	

Note

If you select parameter EINZ = 0 and if you thus have the requested test executed in one PLC cycle, the cycle time can rise so that the S5-95F goes to STOP with timeout. Please observe the execution times of the test components (see Table 9-32).

Execution Times of FB252

Table 9-32. Execution Times of FB252

Test Component	Parameter AUFT	Execution Time of the Test Component when Carried Out ...	
		... in its Entirety	... in Slices
CPU	00_H	37 ms	2 ms
Operating system comparison	01_H	(20 s) It is not possible to process the test component in its entirety via FB252.	4 ms
RAM with STEP 5 objects*	02_H	Approx. 1.4 ms per STEP 5 statement	Processing in slices not possible
Onboard DI	03_H	117 ms	5 ms
Onboard DQ	04_H	63 ms	3.2 ms
Onboard interrupt DI	05_H	60 ms	8.3 ms
Onboard counters	06_H	80 ms for 1 counter, 120 ms for 2 counters	3.8 ms
External I/Os, DI	07_H	400 ms to 700 ms, depending on the bus configuration	30 ms
External I/Os, DQ	08_H	400 ms to 700 ms, depending on the bus configuration	30 ms
Processor RAM	09_H	3.6 ms	3.6 ms
Memory submodule	$0A_H$	1.8 ms	1.8 ms
Power failure routine	$0B_H$	1.9 ms	1.9 ms
Scan time monitor	$0C_H$	800 ms	1.8 ms
Load voltage failure routine	$0D_H$	5.7 ms	5.7 ms
Logical program counter	$0E_H$	1.8 ms	1.8 ms
Short-circuit test*	$0F_H$	On-board DIs per test DQ: 12 ms At least 1 external DI per test DQ: 19 ms	Processing in slices not possible

* The test object is always completely processed, without regard to the EINZ parameter.

Example: Execution of the Short-Circuit Test for I/O Type D or E

Test digital input DI 32.3 via digital output DQ 33.2 for short circuit. You have programmed this circuit with COM 95F. You will find the following entry in DB1:

KT DQ 33.2 32.3; (DQ 33.2 tests DI 32.3)

You store the condition for calling the short-circuit test in flag 10.0. Flag 10.6 = 1 and flag 10.7 = 0.

In DB10 you have stored the parameters for calling FB252 as follows:

DB10	STL	Explanation
0:	KH = 0F00;	Short-circuit test (DL0=$0F_H$)
1:	KH = 0000;	
2:	KM = 0000 0100 0000 0000;	
3:	KH = 0000;	

You call FB252 in OB1. Here in OB1 you have the call JU FB10. In FB10 you have the call for FB252.

FB10		STL	Explanation
	:		
	:	CDB 10	Call DB10
	:	A F 10.0	If F 10.0 = 1 then short-circuit test
	:	JU FB252	Call FB252
NAME	:	AGF: TEST	
AUFT	:	DL 0	DL 0=15 Short-circuit test
TYP	:	DW 1	Irrelevant for short-circuit test
NR	:	DL 2	Short-circuit test DQ
BED	:	F 10.6	Give priority to test (F 10.6 = 1)
EINZ	:	F 10.7	Test in a PLC cycle (F 10.7 = 0)
PAFE	:	DR 3	Message byte

If you call up FB252 as described above, then the short-circuit test is executed for the lines of DI 32.3. As soon as the short-circuit test is terminated, FB252 enters the value 00_H into byte DR3 in DB10.

Notes on the Error Responses during the Short-Circuit Test



Caution

If the S5-95F recognizes an error during the short-circuit test, it reacts with STOP (even if the reaction you have parameterized is for standard value generation with AND, OR or OLD value). Only if you have programmed the signal group to react with passivation will the S5-95F really react with passivation.

Short-Circuit Test with Safety Responsibility



Safety Note

If the short-circuit test is responsible for the safety of the tested sensor line, then you must call FB 252 at least:

- once per process safety time in the case of single-channel, failsafe sensors
- once within the second error occurrence time in the case of two-channel sensors.

Special Characteristic of the DQ Test

During the DQ test, the S5-95F briefly resets the outputs to be tested. If you wish to call up the DQ test by means of function block FB 252, then please refer to section 4.3.

9.2.10 Depassivation Block - FB255 -

With FB255, you can
- re-integrate the I/Os for a faulted signal group into the process, or
- delete the entries in the system event DB (FB254)

for instance after repairing a hardware component in RUN mode.

The reasons for passivation of I/Os or for standardized value generation of I/O signals may be:

- Defective sensors
- Defective or incorrect cabling
- Defective modules

Calling and parameter assignments

Parameter	Meaning	Type	Assignment	STL
AUFT	Type of job	I/KY	0.0: Acknowledge error (delete error blocks in DB 254) 1,X Depassivate signal group with X=0,1,2 to 31 and delete corresponding error block in DB254	: JU FB 255 Name : AGF:DEPA : SIGR : DEPA : PAFE
DEPA	Trigger job	I/BI	Edge change from 0 to 1 triggers FB255	
PAFE	Parameter error	Q/BY	00_H = Test terminated and depassivation successful 11_H = Signal group not passivated 31_H = Error detected in signal group, signal group remains passivated 41_H = Block assigned wrong parameters $D0_H$ = Depassivation active, test active $E0_H$ = Depassivation active, waiting for test	

Depassivation Routine

> **Note**
>
> Call FB255 only in the cyclic program (OB1). Generate an independent routine for depassivation in your control program. In this routine, you make sure that the system cannot assume any impermissible statuses during depassivation of the I/Os. Special attention should be paid to the statuses of the digital outputs.

Special Characteristics of Depassivation

- Please note that all failsafe I/Os are tested when you call up FB255 (not only the I/Os with the signal group to be depassivated).
- ***Calling FB255*** is permissible only if FB252 is ***not active with an I/O test*** (DI, DQ, interrupt DI or counter test). An active test can be interrogated at the PAFE output parameter (PAFE = $D0_H$).
- FB255 must be invoked in the user program until depassivation is completed (evaluate PAFE parameter).
- Depassivation can be accelerated by multiple calling of FB255.
- FB255 does not recognize a discrepancy present at a DI during depassivation. In such a case, the signal group is briefly depassivated and automatically repassivated after elapse of the discrepancy time.

Messages in the PAFE Byte

During the depassivation, the PAFE byte is $D0_H$ or $E0_H$. After successful completion of depassivation, the PAFE parameter is 00_H. If the function block FB255 recognizes an error in the signal group during depassivation, then the PAFE byte is 31_H.

Example for Calling FB255

OB1 STL	Explanation
: : JU FB 255 NAME: AGF: DEPA AUFT: KY 0,0 DEPA: I 1.0 PAFE: FY100	 Call depassivation block Job: Acknowledge error message Trigger depassivation with edge at I 1.0 Reserve FY100 for PAFE

9.3 Parameterizing Internal Functions in DB1

The S5-95F offers functions that you must program according to your requirements. These functions are:

- Using interrupt inputs
- Using counter inputs
- Using the integral real-time clock
- Using failsafe onboard inputs and outputs
- Using failsafe external modules
- Exchanging data via SINEC L1
- Changing call-up interval for time-controlled program processing (OB13) (see Chapter 7)
- Assigning system parameters

> **Note**
>
> For systems requiring approval, you must use the COM 95F software package to parameterize DB1 (see COM 95F manual).

On the following pages, the configuration of DB1 and the meaning of the default parameters are described.

9.3.1 Configuration and Default Settings for DB1

Data block DB1 required for system parameterization is already integrated in the S5-95F programmable controller with preset values (default parameters). If you use failsafe external I/Os, the default parameters are added automatically. DB1 remains valid until you modify it with COM 95F (see COM 95F manual).

After performing an overall reset, you can load the default DB1 from the programmable controller into your programmer and display it on the screen or evaluate it. Figure 9-10 shows the default parameters for an S5-95F programmable controller without failsafe external I/Os.

```
  0:     KC ='DB1 OBHI: HI 59.0 J HI 5';
 12:     KC ='9.1 J HI 59.2 J HI 59.3 ';
 24:     KC ='J ; OBP: EB Y 0 AB 32 0 ';
 36:     KC ='; OBSI: SIN 32.X S N  SI';
 48:     KC ='N 33.X S N  SIN 59.X S N';
 60:     KC ='  ; OBDE: DE 32.X S DE 3';
 72:     KC ='3.X S ; OBC: CAP 0 N    ';
 84:     KC ='  CBP 0 N     ; KT: DA 3';
 96:     KC ='3.X E DA 34.X E ; SIG: S';
108:     KC ='GRP Z S           ; PERI ;
120:     KC =': ST  0 S  ST  1 S ST  2';
132:     KC =' S  ST  3 S ST  4 S  ST ';
144:     KC =' 5 S ST  6 S  ST  7 S ST';
156:     KC ='  8 S  ST  9 S ST 10 S  ';
168:     KC ='ST 11 S ST 12 S  ST 13 S';
180:     KC =' ST 14 S  ST 15 S ST 16 ';
192:     KC ='S  ST 17 S ST 18 S  ST 1';
204:     KC ='9 S ST 20 S  ST 21 S ST ';
216:     KC ='22 S  ST 23 S ST 24 S  S';
228:     KC ='T 25 S ST 26 S  ST 27 S ';
240:     KC ='ST 28 S  ST 29 S ST 30 S';
252:     KC ='  ST 31 S ; EXDE: DE Y.X';
264:     KC =' 1L1  ; EXAE: AE X M 1L1';
276:     KC =' 10A  ; SL1: #PGN N#  SL';
288:     KC ='N N   ; SL1A: KBE N     ';
300:     KC ='KBS N     SF N      EF N';
312:     KC ='      ; SL1B: KBE N     ';
324:     KC ='KBS N     SF N      EF N';
336:     KC ='      ; SL1S: UVB N     ';
348:     KC ='D1S N D1E N D2S N D2E N ';
360:     KC ='SNTS1  0  9 SNTS2  0  9 ';
372:     KC ='SNTE1  0  9 SNTE2  0  9 ';
384:     KC ='TD1S 0  TD2S 0  TD1E 0 S';
396:     KC ='  TD2E 0 S  ; SDP: SYID ';
408:     KC ='0 DBCON 200 251 CYST N  ';
420:     KC ='AGCYC 25  DBPAR N ERSI N';
432:     KC ='  ERCP N  ; TFB: OB13  0';
444:     KC ='  ; CLP: STW N     CLK N';
456:     KC ='     SET N     OHS N    ';
468:     KC =' OHE N     TIS N     STP';
480:     KC =' N     SAV N     CF 0 ; ';
492:     KC ='CRC: CRC !??! ; END ';
```

Figure 9-10. DB1 with Default Parameters

DB1 consists of several parameter blocks which are framed by the start ID "DB1" and the "END" ID. Each parameter block begins with a block ID (shown in Figure 9-10 in the shaded background), followed by a colon. The individual parameters for each function are contained in these parameter blocks.

Table 9-33. Parameter Blocks and Their IDs (S5-95F)

Block ID	Explanation/Default Setting
'DB1 ';	Start ID
'OBHI: ';	Onboard hardware interrupt: Parameter block for asynchronous interrupt processing (see Chapter 12)
'OBP: ';	Onboard I/Os: Parameter block for the redundant onboard inputs and outputs
'OBSI: ';	Onboard software interrupt: Parameter block for synchronous interrupt processing (see Chapter 12)
'OBDE: ';	Onboard digital inputs: Parameter block for the onboard digital inputs
'OBC: ';	Onboard counter: Parameter block for the counter inputs
'KT: ';	Short-circuit test: Parameter block for the short-circuit test DQs
'SIG: ';	Signal groups: Parameter block for signal group parameterization
'PERI: ';	I/Os: Parameter block for external I/O configuration
'EXDE: ';	External digital inputs: Parameter block for the discrepancy time of the external digital inputs
'EXAE: ';	External analog inputs: Reserved parameter block for failsafe analog input
'SL1: ';	SINEC L1: Parameter block for SINEC L1 programmer bus and/or SINEC L1 data bus
'SL1A: ';	SINEC L1 at subunit A: Parameter block for non-failsafe SINEC L1
'SL1B: ';	SINEC L1 at subunit B: Parameter block for non-failsafe SINEC L1
'SL1S: ';	SINEC L1, failsafe: Parameter block for failsafe SINEC L1 at subunit A and/or subunit B
'SDP: ';	System-dependent parameter: Parameter block for system-dependent parameters
'TFB: ';	Timer function block: Parameter block for time-controlled program processing
'CLP: ';	Clock parameters: Parameter block for integral real-time clock
'CRC: ';	CRC checksum: Parameter block for testing the STEP 5 program. Parameter is entered by the operating system.

The sequence of the parameter blocks in DB1 is not fixed. A semicolon (;) must be at the end of each parameter block. At least one filler must be between the semicolon and the next block ID.

9.3.2 How to Assign Parameters in DB1 without COM 95F

Use the following steps to change or expand the preset values of DB1:

Display the default DB1 on the programmer.
Position the cursor on the desired parameter block.
Change or expand the parameters.
(for an explanation and possible parameter values see section 9.3.6)
Transfer the changed DB1 to the programmable controller.
Switch the programmable controller from STOP to RUN.

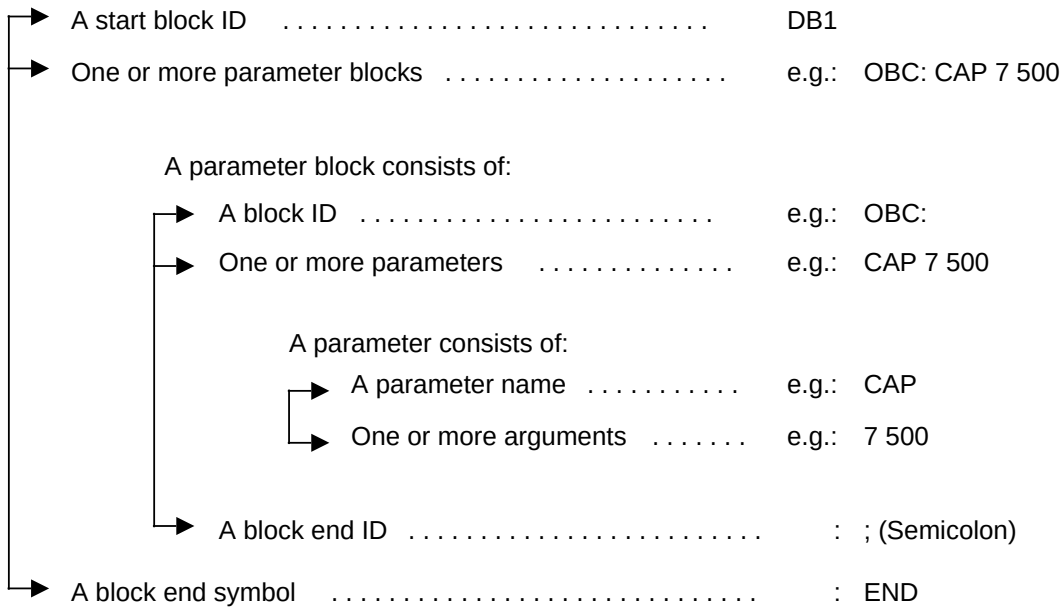
Changed DB1 parameters are accepted. For all settings that are not changed, the default settings remain valid. The default settings are selected so that the S5-95F assumes a status as safe as possible.

Note

If the programmable controller recognizes a parameter error in DB1, the S5-95F enters the cause of the error in the system event DB (error No. 70 "Parameter assignment error in DB1"). The S5-95F does not go to RUN.

9.3.3 Rules for Setting Parameters in DB1

DB1 consists of the following:



In the following section are all of the rules that have to be followed if you want to change or expand entire parameter blocks in DB1. Follow these steps or the programmable controller will not understand what you have entered.

1. Permissible fillers are: Space and comma

2. Start ID "DB1"
   DB1 must begin with the start ID "DB1". Do not separate the three characters from each other. After the start ID, there must be at least one filler.

3. The start ID and filler are followed by the block ID for the parameter block. The sequence of the parameter blocks in DB1 is random. The block ID identifies a block and its corresponding parameter. The block ID "SL1", for example, stands for the SINEC-L1 parameter. You must enter a colon immediately after the block ID. If the colon is missing, then the programmable controller skips this block and displays an error message. You must add at least one filler after the colon of a block ID.

4. The parameter name comes next. Parameter names are names for single parameters within a parameter block. Within a block, the first four characters of a parameter name must be different from each other. After the parameter name, you must add at least one filler.

5. At least one argument is attached to each parameter name. An argument is either a number, a character string or a STEP 5 operand that you must enter. If several arguments belong to a parameter name, then every argument must be followed by at least one filler (even the last one).

6. Use a semicolon (;) to identify a block end. After the semicolon, you must enter at least one filler. Leaving out the semicolon leads to misinterpretation in the programmable controller.

7. After the semicolon, additional parameter blocks can follow. (Use steps 2 through 5 to create additional parameter blocks.)

8. After the end of the last parameter block, you must enter the end ID "END". This identifies the end of DB1. If you forget to enter an end ID, this leads to errors in the programmable controller.

9. The "CRC" block must be present. The S5-95F enters the argument itself.

Steps 1 through 7 present the minimal requirements for setting the parameters. Beyond that, there are additional rules that make it easier for you to assign parameters.

For example:

- You have the ability to add comments.
- You can expand the mnemonics used as parameter names by using plain text.

Comments can be added anywhere a filler is allowed. The comment symbol is the pound (#) sign. The comment symbol must be placed at the beginning and at the end of your comment. The text between two comment symbols may not contain an additional #.

Example: #Comment# .
At least one filler must follow the # sign.

In order to make it easier to read parameter names, you can add as many characters as you wish if you add an underscore (_) after the abbreviated parameter name.

Example: SF becomes SF_SENDMAILBOX .
At the end of the input, you must add at least one filler.

There is a rule of thumb that will help you check DB1. You should include at least one filler in the following instances.

- After the start ID
- Before and after the block ID, parameter name, argument, and semicolon

9.3.4 How to Recognize and Correct Parameter Errors

Should an error occur while assigning parameters and the programmable controller does not go to the ”RUN” mode, you have two possibilities for recognizing errors.

- By using the COM 95F software (see COM 95F manual)
- By direct interpretation of the entry in the system event DB (see section 15.3.9).

9.3.5 Transferring Changed DB1 Parameters to the S5-95F Programmable Controller

DB1 determines to a limited extent the system parameters of the S5-95F. The S5-95F reads DB1 only once during startup.

If you assign new parameters to DB1 (in the test mode), then the S5-95F must read DB1, as otherwise the changed parameters do not become valid. You must therefore carry out a cold restart on the S5-95F.

To carry out a cold restart on the S5-95F and to accept the changed parameters of DB1 in the S5-95F, proceed as follows:

- Switch from RUN to STOP and again to RUN.
- Switch the power off and then on or

If you do not assign parameters in DB1, then the S5-95F generates a default DB1, where it also enters the current I/O configuration. The default settings of DB1 are selected so that the S5-95F can assume a status as safe as possible. With this default DB1, the S5-95F goes to RUN.

If you assign parameters in DB1, then the S5-95F checks this DB1. If the S5-95F does not recognize any syntax errors in your DB1, it goes to RUN.

If the S5-95F recognizes an error in your DB1 (parameterization error), then the S5-95F goes to STOP and writes a corresponding error message to the system event data block DB254 (see Chapter 15).

9.3.6 Reference Guide for Setting Parameters in DB1

The following table gives you an overview of possible DB1 parameters. You require this table if you wish to evaluate the parameters entered on the programmer without using COM 95F.

Parameter	Argument	Explanation
Block ID: OBHI		Onboard hardware interrupt
Default setting		
HI HI HI HI	59.0 J 59.1 J 59.2 J 59.3 J	The four onboard interrupt inputs I 59.0 to 59.3 are enabled for OB2 interrupt processing.
Permissible changes		
HI HI HI HI	59.0 N 59.1 N 59.2 N 59.3 N	Each onboard input of IB 59 can be disabled for OB2 interrupt processing. Therefore change the argument "J" (yes) from the default setting to "N" (no). Note: Inputs of IB 59 that you disable for OB2 interrupt processing can be used for OB3 interrupt processing (see parameter block OBSI:). Inputs that you use neither for OB2 nor for OB3 interrupt processing can be used as failsafe DIs with short discrepancy time.
Block ID: OBP		On-board I/Os
Default setting		
EB AB	Y 0 32 0	All redundantly operated onboard inputs (IB 32, 33 and 59) are assigned to signal group 0. All redundantly operated onboard outputs (QB 32) are assigned to signal group 0.
Permissible changes		
EB AB	y s 32 s	You can assign different signal group(s) to each input or output byte: permissible range of values byte no. y: y = 32, 33, 59, Y (Y: all) signal group s: s = 0, 1, ... 31 Note: For input and output bytes that are not changed, the default setting remains valid.

Parameter	Argument	Explanation
Block ID: OBSI		Onboard software interrupt
Default setting		
SIN SIN SIN	32.X S N 33.X S N 59.X S N	OB3 interrupt processing with negative edge at input. The short discrepancy time (S) applies for the interrupt inputs. The inputs are, however, not enabled (N).
Permissible changes		
SIN SIN SIN SIP SIP SIP SIPN SIPN SIPN SINP SINP SINP	32.x t w 33.x t w 59.x t w 32.x t w 33.x t w 59.x t w 32.x t w 33.x t w 59.x t w 32.x t w 33.x t w 59.x t w	OB3 interrupt processing with negative edge at input. OB3 interrupt processing with positive edge at input. OB3 interrupt processing with positive or negative edge at input. Permissible range of values Bit no. x: x = 0, 1, ... 7, X (X: all) for bytes 32, 33 x = 0, 1, ... 3, X (X: all) for byte 59 Discrepancy time t: t = S Short discrepancy time (approx. 1 ms) t = M Medium discrepancy time (approx. 5 ms) t = 1 ... 255L1: Discrepancy time in OB1 cycles Enabling w: w = J (YES): Inputs are enabled w = N (NO): Inputs are not enabled

Parameter	Argument	Explanation
Block ID: OBDE		Discrepancy time for onboard inputs
Default setting		
DE DE	32.X S 33.X S	All (non-interrupt) onboard inputs have the short discrepancy time as default setting.
Permissible changes		
DE DE	32.x t 33.x t	The discrepancy time can be changed for each onboard input. Permissible range of values Bit no. x: x = 0, 1, ... 7, X (X: all bits) Discrepancy time t: t = S Short discrepancy time t = 1 ... 255L1: Discrepancy time in OB1 cycles t = 1 ... 255L13: Discrepancy time in OB13 cycles

Parameter	Argument	Explanation
Block ID: OBC		Onboard counter
Default setting		
CAP CBP	0 N 0 N	Counters register pulses with positive edge at the counter input. Counters are assigned to signal group 0 and not yet enabled.
Permissible changes		
		Counters are enabled by parameterizing of a signal group and a comparison value.
CAP CBP	s p s p	Counters register pulses with positive edge.
CAN CBN	s p s p	Counters register pulses with negative edge.
CCP	s q	Counters A&B are cascaded. Cascaded counter registers pulses with positive edge.
CCN	s q	Counters A&B are cascaded. Cascaded counter registers pulses with negative edge.
		Permissible range of values Signal group s: s= 0, 1, ... , 31 Comparison value p: p=0*, 1, ... , 65536 Comparison value q: q=0*, 1, ... , 4294967296 * If p or q = 0, the maximum comparison value is loaded
Block ID: KT		Onboard outputs for short-circuit test
Default setting		
DA DA	33.X E 34.X E	The onboard outputs DQ 33.0 to 33.3 and DQ 34.0 to 34.3 are not used for short circuit test and can be used as non-failsafe outputs.
Permissible changes		
DA DA	33.x p_i 34.x p_i	DI bits pi are assigned to short-circuit test DQ 33.x or 34.x Bit no. of DQ 33/34.x: x = 0, 1, 2, 3 Addresses of DI: P_i=a.b a=0 ... 33 a=59 b=0 ... 7 b=0 ... 3 Pi = E: End of list Note: You must assign DIs endangered by short circuit to different short-circuit test DQs. A maximim of 8 DIs can be assigned to one test DQ. Example: DQ 34.0 0.4 32.0 33.7 E

Parameter	Argument	Explanation
Block ID: SIG		**Define signal groups**
Default setting		
SGRP	Z S	When an error is recognized, all signal groups (0 to 31) respond with PLC STOP
Permissible changes		
SGRP	z v	You can assign each signal group its own error response. Permissible range of values Signal group z: z = 0, 1, ..., 31, Z (Z: all signal groups) Response in the case of error v: v = S, P, L, O, A S = PLC STOP P = Passivation L = Read in previous value O = ORing of subunit signals A = ANDing of subunit signals
Block ID: PERI		Slots of the external I/O modules
Default setting		
ST ⋮ ST	0 S 31 S	All slots for external I/O modules are reserved for single-channel (non-failsafe) modules.
Permissible changes		
ST	xRs t	Slot is occupied with failsafe module. Permissible range of values Slot no. in subunit A: x = 0, 2, ... 30 Redundantly operated module belongs to signal group s s = 0, 1, ... 31 I/O type t t = DI, DQ Note: The slot no. (x + 1) in subunit B must not be parameterized.

Parameter	Argument	Explanation
Block ID: EXDE		Discrepancy time for external DI
Default setting		
DE	Y.X 1L1	Discrepancy time for all redundant external DIs: one OB1 cycle
Permissible changes		
DE	x.y t	You can assign individual bits or bytes a different discrepancy time Byte address x: x = 0, 2, 4, ..., 30, X (X: all) Bit no. y: y = 0, 1, ..., 7, Y (Y: all) Discrepancy time t: t = 1 to 255L1 Discrepancy time in OB1 cycles t = 1 to 255L13 Discrepancy time in OB13 cycles
Block ID: SL1		SINEC L1 basic parameter assignment
Default setting		
PGN SLN	N N	SINEC L1 programmer bus not used SINEC L1 data bus not used
Permissible changes		
PGN SLN	x y	If you use the SINEC L1 programmer bus and/or the SINEC L1 data bus, then enter the number of the node after the Node no. at the programmer bus x: x=1, 2, to 30 Node no. at the data bus y: y=1, 2, to 30
Block ID: SL1A		Non-failsafe SINEC L1 at subunit A
Default setting		
KBE KBS SF EF	N N N N	Coordination bytes, send and receive mailboxes are not assigned default parameters.
Permissible changes		
KBE KBS SF EF 	MBx DByDWx MBx DByDWx MBx DByDWx MBx DByDWx	If you use a SINEC L1 data bus, you must define the location of the coordination bytes and the send and receive mailboxes. Permissible range of values No. of the flag byte/data word x: x=0,1 to 255 Block no. of DB y: y=2, 3, to 251

Parameter	Argument	Explanation
Block ID: SL1B		Non-failsafe SINEC L1 at subunit B
Default setting		Coordination bytes, send and receive mailboxes are not assigned default parameters.
KBE	N	
KBS	N	
SF	N	
EF	N	
Permissible changes		If you use a SINEC L1 data bus, you must define the location of the coordination byte and the send and receive mailboxes. Permissible range of values No. of flag byte/data word x: x = 0, 1, to 255 Block no. of DB y: y = 2, 3, to 251
KBE	MBx DByDWx	
KBS	MBx DByDWx	
SF	MBx DByDWx	
EF	MBx DByDWx	

Parameter	Argument	Explanation
Block ID: SL1S		Failsafe SINEC L1 Parameters
Default setting		The failsafe SINEC L1 data bus is not activated.
UVB	N	User valid bit
D1S	N	Data path 1, send
D1E	N	Data path 1, receive
D2S	N	Data path 2, send
D2E	N	Data path 2, receive
SNTS1	0 9	Slave number and mode of destination slave for data path 1
SNTS2	0 9	Slave number and mode of destination slave for data path 2
SNTE1	0 9	Slave number and mode of source slave for data path 1
SNTE2	0 9	Slave number and mode of source slave for data path 2
TD1S	0	Send safety time for data path 1
TD2S	0	Send safety time for data path 2
TD1E	0 S	Receive safety time for data path 1 and STOP reaction
TD2E	0 S	Receive safety time for data path 2 and STOP reaction
Permissible changes		
UVB	MBx	You can install the failsafe SINEC L1 data bus at subunit A, subunit B or even at both subunits (fault-tolerant). Permissible range of values Byte no. x: x = 0, 1, ..., 255 Connected subunit k: k = N (none) A (subunit A) B (subunit B) H (subunits A&B) Node no. of slave n: n = 1 to 30; with SNTS1, n = 31 also possible (Broadcast) Slave type t: t=1 (115F-14 mode), 5 (115F-15 mode), 9 (95F mode) Safety time u: u = 0, 3, ..., 1638 (times 100 ms) Response in case of SINEC L1 failure, p: p = S (PLC STOP) P (passivation of data paths with response in STEP 5 program)
D1S	k	
D1E	k	
D2S	k	
D2E	k	
SNTS1	n t	
SNTS2	n t	
SNTE1	n t	
SNTE2	n t	
TD1S	u	
TD2S	u	
TD1E	u p	
TD2E	u p	

Parameter	Argument	Explanation
Block ID: SDP		System-dependent parameter
Default setting		
SYID	0	System identification number (here 0)
DBCON	200 251	Data blocks with constant contents (here DB200 to DB251) DBs with constant contents must not be changed in RUN mode
		PLC cycle statistics
CYST	N	Scan time monitoring through software watchdog
AGCYC	25	(here 25 x 10 ms = 250 ms)
		Block no. of parameter control DB
DBPAR	N	Transferring of system messages to SINEC L1 master
ERSI	N	Transferring of system messages to CP 521
ERCP	N	
Permissible changes		
SYID	y	You can change the system-dependent parameters
DBCON	a b	Permissible range of values:
CYST	w	freely available numbers to identify the system y:
AGCYC	t	y = 0, 1, ..., 255
		Interval a, b for data blocks with constant contents
DBPAR	z	a or b = 2, ..., 251, where a < b. For data blocks with no
ERSI	w	constant contents, you must set a = B.
ERCP	p	Maximum PLC cycle time t:
		t = 10, 11, ..., 255 (times 10 ms)
		Block no. of the parameter control DB z:
		z = 2, 3, ..., 253, N N means no parameter control DB
		Slot of CP 521:
		p = 0, 1, ..., 7, N N means no CP 521 used
		Enabling w:
		w = J (YES): Functions are enabled
		w = N (NO): Functions are not enabled
Block ID: TFB		Time-controlled program processing (timer function block)
Default setting		
OB13	0	Time-controlled program processing not activated
Permissible changes		
OB13	x	By specifying an argument you activate the time-controlled program processing. Permissible range of values x = 0, 1, 2, ... 655350 (times 1 ms) Entry is possible in steps of 1 ms. For the call you must use only multiples of 10 ms (otherwise, the system would round the value entered). Enter call intervals T of OB13 in units of 10 ms in the system data SD97 (T = 1, 2, ..., 65535 x 10 ms)

Parameter	Argument	Explanation
Block ID: CLP		Integral real-time clock (clock parameter)
Default setting		Integral real-time clock is not activated
STW	N	Location of the status word (**st**atus **w**ord)
CLK	N	Location of the clock data (**cl**oc**k** data)
OHS	N	Set operating hours counter (**o**perating **h**ours counter **s**et)
OHE	N	Enable the operating hours counter (**o**peration **h**our counter **e**nable)
SET	N	Set date/time
TIS	N	Set prompting time
STP	N	Update clock during PLC STOP
SAV	N	Save clock time after the last change from RUN to STOP or POWER OFF
CF	0	Correction factor
Permissible changes		
STW	DBxDWy	Location of the status word in the parameter control DB
CLK	DBxDWy	1st word of clock data in the parameter control DB
SET	wt dd.mm.yy[1] hh:mn:ss[1] AM/PM[2]	Set date/time
OHS	hhhhhh:mn:ss[1]	Set operating hours counter (**o**peration **c**ounter **s**et)
OHE	Y/N	Enable operating hours counter (**o**peration **h**our counter **e**nable)
TIS	wt dd.mm hh:mn:ss[1] AM/PM[2]	Set prompting time (**t**imer **i**nterrupt **s**et)
STP	w	Update clock during STOP (**st**o**p**)
SAV	w	Save clock time after last change from RUN to STOP or POWER OFF (**sav**e)
CF	p	Enter correction factor (**c**orrection **f**actor)

Weekday (Sun - Sat)	wd = 0 to 7	Seconds	ss = 00 to 59
Day	dd = 01 to 31	Correction factor	p = -400 to 400
Month	mm = 01 to 12	DB number	x = 2 to 251
Year	yy = 0 to 99	Byte number	y = 0 to 255
Hours	hh = 00 to 23	Word number	z = 0 to 254
Minutes	mn = 00 to 59	Enable	w = Y (YES)/N (NO)

Block ID: CRC		Checksum
CRC	!??!	Checksum The checksum is stored by the operating system. Only parameter CRC: must be reserved.

[1] If an argument (for example seconds) need not be updated, simply enter XX! The clock continues to run with the current data. This argument is not taken into account in the TIS parameter block.

[2] If you input AM or PM after the clock time, the clock runs in the 12-hour mode. If you omit this argument, the clock runs in the 24-hour mode. You must use the same time mode in the SET and TIS parameter blocks. When assigning the parameters for the integral real-time clock, please also refer to Chapter 10 "Integral Real-Time Clock".

10 The Integral Real-Time Clock

10.1	Operating Principle and Parameterization of the Integral Real-Time Clock	10- 1
10.1.1	Setting the Clock Parameters in DB1	10- 1
10.1.2	Operating Principle of the Clock	10- 1
10.1.3	Transfer and Battery Backup of Clock Parameters	10- 2
10.1.4	Entering the Clock Time Correction Factor	10- 2
10.2	Structure of the Clock Data Area	10- 3
10.3	Structure of the Status Word and How to Scan it	10- 6
10.4	Using the Programmer to Read and Set the Integral Real-Time Clock	10- 9
10.5	Programming the Integral Real-Time Clock in the Control Program	10- 10
10.5.1	Reading and Setting the Clock	10- 10
10.5.2	Storing the Updated Time/Date after a RUN to STOP or RUN to POWER OFF Transition	10- 14
10.5.3	Setting the Prompt Time	10- 14
10.5.4	Setting the Operating Hours Counter	10- 18

Figures

10-1 Accessing the Clock Data Area 10- 3

Tables

10-1 Clock Data in the Clock Data Area 10- 4
10-2 Range Definitions for Clock Data 10- 5
10-3 Significance of Bits 0, 1 and 2 of the Status Word 10- 7
10-4 Significance of Bits 4 and 5 of the Status Word 10- 7
10-5 Significance of the Operating Hours Counter Flag (Bits 8, 9 and 10 of the Status Word) 10- 8
10-6 Significance of the Prompt Time Flag (Bits 12, 13 and 14 of the Status Word) 10- 8
10-7 Example: Setting the Clock Time via the Programmer Function "BLOCK OUTPUT" 10- 9

10 The Integral Real-Time Clock

The S5-95F is provided with an integral real-time clock, which offers you the possibility of controlling the process sequences. The clock data must, however, not be used to initiate safety functions.

10.1 Operating Principle and Parameterization of the Integral Real-Time Clock

The integral real-time clock offers you the following functions:

- Clock and calendar function
  Used to configure clock-time-dependent control, for example
- Prompt and alarm function
  Used to monitor the duration of a process, for example
- Operating hours counter
  Used to monitor inspection intervals, for example

10.1.1 Setting the Clock Parameters in DB1

In the default setting, the clock is not activated. If you wish to use it, you must set the clock parameters in DB1.

There are two possibilities to parameterize the integral real-time clock:

- Direct setting of clock parameters in DB1 (see section 9.3.6)
- Using COM 95F to set clock parameters (see COM 95F manual)

The integral clock requires a clock data area and a status word. We recommend you to store the clock data area and the status word in the parameter control DB so that you can also set the clock in the safety mode.

10.1.2 Operating Principle of the Clock

Operation of the integral real-time clock is independent of the operating state set.

Data exchange between the integral real-time clock and the control program always goes through the clock data area. The clock stores current values for time, date and operating hours counter in the clock data area. You can transfer into the clock data area the values for the time, date, prompt time, and operating hours counter that you want the clock to use.

You can scan the status word to identify setting errors, for example. Or you can change certain status word bits to deliberately disable or enable transfer or read operations.

Refer to sections 10.2 and 10.3 for detailed information on the clock data area and the status word.

10.1.3 Transfer and Battery Backup of Clock Parameters

Transfer of Clock Parameters from DB1

The clock parameters are transferred from DB1 to the S5-95F only if you have carried out a ***manual*** S5-95F overall reset in advance. The clock parameters are transferred after the first transition from STOP to RUN. All other transitions from STOP to RUN do not affect the operation of the clock.

Setting the Clock via the Programmer or the Control Program

You can set the clock via the programmer or the control program (see sections 10.4 and 10.5). We recommend to store the clock data area and the status word in the parameter control DB so that you can also set the clock in the safety mode.

Battery Backup and Replacement of Battery

During POWER OFF, the integral real-time clock is supplied by the backup battery; the clock therefore continues running both during STOP and POWER OFF.

Change the battery only during POWER ON, as otherwise clock data can be lost. Observe the general rules for battery replacement (see Chapter 2).

10.1.4 Entering the Clock Time Correction Factor

The exactness of the clock is temperature-dependent. You can configure a correction value to increase the clock's exactness. The correction value is output in s/month. You must measure how many seconds per month the clock runs fast or slow. A month is defined as 30 days.

Example: Your measurements indicate the clock is 12 s too slow in 4 days. That would be 90 s too slow in 30 days. The correction value is +90 s/month.

Enter the correction factor with COM 95F or edit the correction factor directly in DB1.

Note

If you set a correction factor other than zero, the system may fail to perform an automatic warm restart following an extended power outage (Error message 7: Hardware fault in internal RAM).

10.2 Structure of the Clock Data Area

Convention:

You must store the location of the clock data area in DB1. To enable you to set the clock also in the safety mode via the programmer, we recommend you to store the clock data area and the status word in the parameter control DB.

For reasons of clarity, we assume on the following pages that the clock data area occupies data words DW 0 to DW 21 in DB200 and that the status word occupies the data word DW 22.

Structure of the Clock Data Area

Data exchange between DB1 or the control program and the integral real-time clock is always through the clock data area. The integral real-time clock stores current time, date and operating hours counter values in the clock data area. DB1 and the control program store the settings for prompt times and operating hours counter in the same data area. Figure 10-1 illustrates the relationship between DB1 or the control program, the clock data area, and the integral real-time clock.

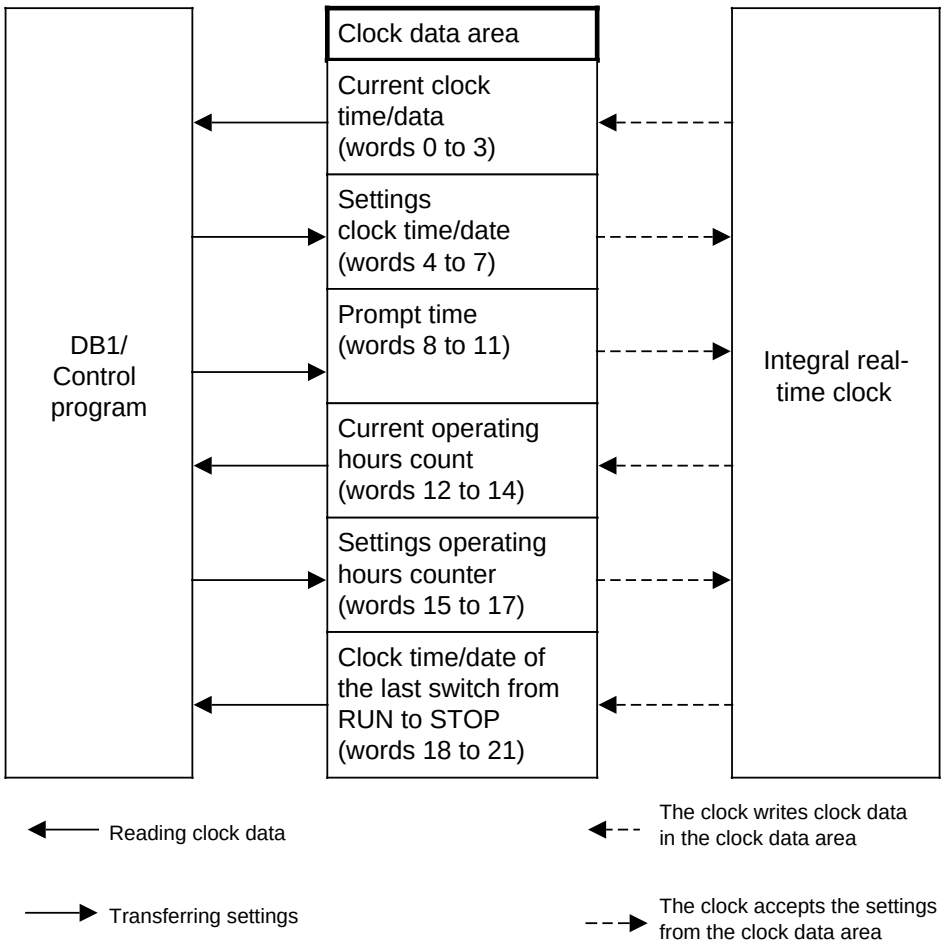


Figure 10-1. Accessing the Clock Data Area

When you set the clock, you have to transfer only the data needed to implement a particular function. For example, if you want to change only the clock function data, you do not have to enter data for the time prompt function or for the operating hours counter.

Table 10-1 provides you with information about where specific clock data is located within the clock data area. The explanations for Table 10-1 follow the table.

Table 10-1. Clock Data in the Clock Data Area

Clock Data Area (Data Word)	Meaning	Left Flag Word	Right Flag Word
0	Current clock time/date	---	Weekday
1		Day	Month
2		Year	AM/PM (Bit 7)*, hour
3		Minute	Second
4	Settings for clock time/date	---	Weekday
5		Day	Month
6		Year	AM/PM (Bit 7)*, hour
7		Minute	Second
8	Time prompt	---	Weekday
9		Day	Month
10		---	AM/PM (Bit 7)*, hour
11		Minute	Second
12	Current operating hours	---	Seconds
13		Minutes	Hours
14		Hours · 100	Hours · 10,000
15	Settings for operating hours counter	---	Seconds
16		Minutes	Hours
17		Hours · 100	Hours · 10,000
18	Clock time/date after a switch from RUN to STOP or POWER OFF (only if bit 5 in the status word = 1)	---	Weekday
19		Day	Month
20		Year	AM/PM (Bit 7)*, hour
21		Minute	Second

* Significant only in the 12-hour mode Bit 7=1 means PM, bit 7 =0 means AM

Make certain you are aware of the following points when you make inputs into the clock data area.

- Entries into the clock data area must be in BCD code.
- The clock settings you enter must be within the range defined in Table 10-2.

Table 10-2. Range Definitions for Clock Data

Variable	Permissible Parameters	Variable	Permissible Parameters
Seconds	0 to 59	Day	1 to 31
Minutes	0 to 59	Month	1 to 12
Hours	In the 24-hour mode: 0 to 23 In the 12-hour mode: for AM: 1 to 12 (12 = 12 o'clock noon) for PM: 81 to 92 (81 = 1 o'clock PM) 0 to 999999 when entering the operating hours	Year	0 to 99
Weekday	1 to 7 1=Sunday 2=Monday 3=Tuesday 4=Wednesday 5=Thursday 6=Friday 7=Saturday		

- If you do not wish to modify one of the setting values, you must enter FFH for this value.

- Entries outside the defined range cause the operating system to set the corresponding error bit in the status word. The operating system resets these error bits in the status word the next time you set the clock if the new settings are within the defined range.

- The AM/PM flag (0 = AM; 1 = PM) is relevant only in the 12-hour mode.
  You must, however, observe the following: Whether the clock is to operate in the 12-hour or 24-hour mode is specified in the status word (see section 10.3).

  If you operate the integral real-time clock in the 24-hour mode, then setting of the AM/PM flag results in setting of the error bit in the status word.

- Please note that you can define the location of the clock data area and that the word numbers indicated in Table 10-1 are only relative numbers.

  If your clock data area lies in a data block and does not start with DW 0, but with DW x, you must add the value x to the word number in Table 10-1.

10.3 Structure of the Status Word and How to Scan it

You can scan the status word to identify errors in the entered settings. You can deliberately change certain bits in the status word to enable or disable transfer or read operations.

You can use designated flag bits to govern the clock's behavior when the programmable controller is switched from the RUN to the STOP mode or during Power OFF.

- To enable you to set the clock also in the safety mode, we recommend you to store the clock data area and the status word in the parameter control DB.

- Access to the clock data area depends on the set operating mode and the signal states of bits 4 and 5 in the status word. You can set or reset these bits using the "S" or "R" operations in the control program.

- The operating system resets the "transfer settings" bits (bits 2, 10, and 14 in the status word) under the following conditions.
  - The settings have been transferred.
  - The settings have not been transferred because they were outside of the permissible range. The corresponding error bits (bits 0, 8, and 12 in the status word) are set.

- The operating system does not reset the "transfer settings" bits (bits 2, 10 and 14 in the status word) under the following conditions.
  - The system data for the clock is either incorrect or not available.
  - The clock data area is too small.
  - The clock is defective (hardware error).

- There are four types of bits in the status word.
  - Clock flags
  - Operating system flags
  - Operating hours counter flags
  - Prompt time flags

Tables 10-3 through 10-6 provide you with information about the significance of the signal states of the respective flags.

Clock Flags

Table 10-3. Significance of Bits 0, 1 and 2 of the Status Word

Bit Number	Signal State	Meaning
0	1	Error in setting entry
	0	No error in setting entry
1	1	12-hour clock mode
	0	24-hour clock mode
2	1	Transfer settings
	0	Do not transfer settings

Operating System Flags

Table 10-4. Significance of Bits 4 and 5 of the Status Word

Operating Mode	Bit Number Status Word	Signal State	Meaning
STOP	4	1	The clock updates only words 0 to 3 (current time/date) in the clock data area.
		0	The clock does not update the clock data area. Words 0 to 3 contain the time at which the last switch from RUN to STOP occurred.
	5	1	Words 18 to 21 contain the time at which the last RUN to STOP switch occurred or the time at which the last Power OFF occurred if bit 4 is also set.
		0	Words 18 to 21 are not used.
RUN	4	1/0	The clock continually updates the clock data area (Words 0 to 17).
	5	1	Words 18 to 21 contain the time at which the last switch from RUN to STOP occurred or the time at which the last Power OFF occurred.
		0	Words 18 to 21 are not used.

Operating Hours Counter Flags

Table 10-5. Significance of the Operating Hours Counter Flag (Bits 8, 9, and 10 of the Status Word)

Bit Number	Signal State	Meaning
8	1	Error in setting entry
	0	No error in setting entry
9	1	Enable the operating hours counter
	0	Disable the operating hours counter
10	1	Transfer the settings for the operating hours counter
	0	Do not transfer the settings for the operating hours counter

Prompt Time Flags

Table 10-6. Significance of the Prompt Time Flag (Bits 12, 13, and 14 of the Status Word)

Bit Number	Signal State	Meaning
12	1	Error in setting entry
	0	No error in setting entry
13	1	The set prompting time is reached
	0	The set prompting time is not reached
14	1	Transfer the settings for the prompt time
	0	Do not transfer the settings for the prompt time

The operating system requires bit numbers 3, 6, 7, 11, and 15. You can not use these bits.

Scanning the Status Word

In a data block, you can use the "PD <data word number> <bit number>" operation to scan the individual bits of a data word.

Example: The status word is stored in DW22. You are checking to see if the set prompt time has been reached.
You program the scan with the operation "PD 22.13".

10.4 Using the Programmer to Read and Set the Integral Real-Time Clock

The following section describes reading and setting of the integral real-time clock via the programmer with the "Block Output" function. In the following, we assume that the ***clock data area and the status word*** have been stored in the ***parameter control DB*** and that data block DB1 has been parameterized correspondingly. When parameterizing the clock using COM 95F, be sure to acknowledge the entry ”Update in STOP” with ”Yes”.

Please note that in the safety mode you can set the integral real-time clock via the programmer only when the PLC is at a STOP.

Reading and Setting the Clock Time

Read the parameter control DB (here DB200) from the S5-95F using the programmer function "BLOCK OUTPUT".
Enter the new settings in DW 4 through DW 7.
If you do not wish to transfer one of the settings, you must mark the corresponding byte with "FF_H". When you then set the clock, the present value is retained.
Set bit 2 of the status word; the settings from DW 4 through DW 7 are transferred to the clock
Impermissible settings are indicated by a bit which is then set in the status word. The clock continues running with the old values.

Table 10-7. Example: Setting the Clock Time via the Programmer Function "BLOCK OUTPUT"

Operand	Format	Explanation
DB 200		Read current clock data
DW 0	KH = 0003	Tuesday
DW 1	KH = 0510	October 5
DW 2	KH = 9312	1993
DW 3	KH = 0000	12 o'clock
		Enter new settings
DW 4	KH = 0002	Monday
DW 5	KH = 0612	December 6
DW 6	KH = 9310	1993
DW 7	KH = 3500	10.35 Uhr
DW 22	KM = 00000000 00000100	If you **set bit 2 in the status word** to "1", the new settings are transferred to the clock.

Terminate the parameter control DB.
After transferring of the new settings, the S5-95F resets bit 2 of the status word.

Reading and Setting the Prompt Time and the Operating Hours Counter

You can change the settings for the prompt function and the operating hours counter in the same way as described above for the clock time settings.

> **Note**
>
> In the test mode, you can also read and set the integral real-time clock via the programmer function "FORCE VAR". In the safety mode, you can use "FORCE VAR" only to read the clock.

10.5 Programming the Integral Real-Time Clock in the Control Program

In the following sections, we use examples to describe how you can set and evaluate the integral real-time clock via the control program.

Convention:

In the following examples, we assume that the parameter set for the clock has already been stored in DB1; the clock data area occupies data words DW 0 through DW 21 in DB200, and the status word occupies data word DW 22.

10.5.1 Reading and Setting the Clock

Example: Setting the Clock Time and Date

Transfer of the settings for the time and date is triggered by input I 0.0. The settings are available in flag bytes FY121 to 127; they have been received, for example, by the SINEC L1 master. Values that you do not want to change must be preset with FF_H.

Input 1.0 defines the clock mode (0 = 24-hour mode, 1 = 12-hour mode). Input 1.1 is the AM/PM bit for the 12-hour mode.

The clock data area is in DB200, beginning with DW0; the status word is DW 22.

OB1 STL	Explanation
:	================================
:	SETTING THE TIME AND DATE
:A I 0.0	================================
:AN F 0.0	FIRST TRANSFER TIME AND DATE VALUES
:S F 20.0	TO FY121 to FY127.
:A I 0.0	CLOCK SETTING TRIGGERED BY
:= F 0.0	SETTING OF F 20.0
:	(IS RESET IN FB 10)
:JU FB 10	
NAME :SET CLOCK	
WDAY : FB 121	WEEKDAY
DAY : FB 122	DAY
MON : FB 123	MONTH
YEAR : FB 124	YEAR
HOUR : FB 125	HOUR
AMPM : I 1.1	AMPM BIT (ONLY RELEVANT IN 12-HOUR MODE)
MIN : FY 126	MINUTES
SEC : FY 127	SECONDS
EER : F 12.1	ERROR BIT
MODE : I 1.0	CLOCK MODE
:BE	

FB10 STL	Explanation
NAME :SET CLOCK	SETTING THE CLOCK
DES :WDAY I/Q/D/B/T/C: I BI/BY/W/D: BY	
DES :DAY I/Q/D/B/T/C: I BI/BY/W/D: BY	
DES :MON I/Q/D/B/T/C: I BI/BY/W/D: BY	
DES :YEAR I/Q/D/B/T/C: I BI/BY/W/D: BY	
DES :HOUR I/Q/D/B/T/C: I BI/BY/W/D: BY	
DES :AMPM I/Q/D/B/T/C: I BI/BY/W/D: BI	
DES :MIN I/Q/D/B/T/C: I BI/BY/W/D: BY	
DES :SEC I/Q/D/B/T/C: I BI/BY/W/D: BY	
DES :ERR I/Q/D/B/T/C: Q BI/BY/W/D: BI	
DES :MODE I/Q/D/B/T/C: I BI/BY/W/D: BI	
:C DB200	OPEN PARAMETER CONTROL DB
:AN =MODE	24H-MODE = 0, 12H-MODE = 1
:JU =CONT	
:SU D 22.1	CLOCK MODE STATUS WORD BIT 1
:A F 20.0	START MONITORING TIME
:L KT 200.0	
:SE T 10	
WEIT :AN F 20.0	FLAG IS RESET IF SETTING
:JC =M001	ALREADY READ INTO CLOCK
:R F 20.0	DATA AREA
:	
:	CLOCK DATA AREA
:L =WDAY	STORE VALUE FOR WEEKDAY
:T DR 4	
:L =DAY	VALUE FOR DAY
:T DL 5	
:L =MON	VALUE FOR MONTH
:T DR 5	
:L =YEAR	VALUE FOR YEAR
:T DL 6	
:L =HOUR	VALUE FOR HOUR
:ON =AMPM	IF 12-HOUR MODE IS SET, AND
:ON =MODE	AM/PM BIT = 1 (AFTERNOON),
:JC =MORN	THE RELEVANT BIT IN
:L KH 0080	THE CLOCK DATA AREA
:OW	IS SET
MORN :T DR 6	
:L =MIN	VALUE FOR MINUTES
:T DL 7	
:L =SEC	VALUE FOR SECONDS
:T DR 7	
:SU D 22.2	TRANSFER SETTINGS
:	(STATUS WORD IS DW 22)
M001 :A T 10	BEC, IF MONITORING
:BEC	TIME NOT YET ELAPSED

FB10 STL (Continued)	Explanation
:TBN D 22.2 :JC =M002 :S =ERR :BEU M002 :TBN D 22.0 :RB =ERR :BEC :S =ERR :BE	HAVE SETTINGS BEEN TRANSFERRED? IF YES, JUMP TO M002 SET ERROR BIT IF THERE ARE ERRORS WERE THERE ERRORS WHILE ENTERING SETTINGS? IF NO, RESET ERROR BIT IF NO ERROR, THEN BEC IF AN ERROR, SET ERROR BIT

Example: Reading the current time and the current date

The time is stored in flag bytes FY30 to FY36, depending on an external event, simulated here by a positive edge at input I 0.5. Flag F 13.1 indicates which mode the clock is operating in. Flag F 13.0 is the AM/PM bit in the 12-hour mode.

The clock data area is in DB200 beginning with DW0, and the status word is DW 22.

OB1 STL	Explanation
: : : :A I 0.5 :AN F 0.1 := F 0.0 :A I 32.5 := F 0.1 : :A F 0.0 :JC FB 13 NAME :READ CLOCK WDAY : FY 30 DAY : FY 31 MON : FY 32 YEAR : FY 33 HOUR : FY 34 AMPM : F 13.0 MIN : FY 35 SEC : FY 36 MODE : F 13.1 :BE	================================ READING TIME AND DATE ================================ TIME AND DATE ARE STORED IN FY30 TO FY36 IN CASE OF A POSITIVE EDGE AT I 32.5. (EXTERNAL EVENT) EDGE TRIGGER FLAG (READING TIME AND DATE) WEEKDAY DAY MONTH YEAR HOUR F 13.0 = 1, AFTERNOON IN 12H-MODE MINUTES SECONDS F 13.1 = 1, IN 12-HOUR MODE

FB13 STL	Explanation
NAME :READ CLOCK	READING THE CLOCK
DES :WDAY I/Q/D/B/T/C: Q BI/BY/W/D/:BY	
DES :DAY I/Q/D/B/T/C: Q BI/BY/W/D/:BY	
DES :MON I/Q/D/B/T/C: Q BI/BY/W/D/:BY	
DES :YEAR I/Q/D/B/T/C: Q BI/BY/W/D/:BY	
DES :HOUR I/Q/D/B/T/C: Q BI/BY/W/D/:BY	
DES :AMPM I/Q/D/B/T/C: Q BI/BY/W/D/:BI	
DES :MIN I/Q/D/B/T/C: Q BI/BY/W/D/:BY	
DES :SEC I/Q/D/B/T/C: Q BI/BY/W/D/:BY	
DES :MODE I/Q/D/B/T/C: Q BI/BY/W/D/:BI	
:C DB 200	
:L DR 0	WEEKDAY
:T =WDAY	
:L DL 1	DAY
:T =DAY	
:L DR 1	MONTH
:T =MON	
:L DL 2	YEAR
:T =YEAR	
:L DR 2	HOUR
:L KH 007F	ERASE AM/PM BIT
:AW	(ONLY RELEVANT IN 12-HOUR MODE)
:T =HOUR	
:TB D 2.7	DISPLAY AM/PM BIT
:= =AMPM	(ONLY RELEVANT IN 12-HOUR MODE)
:L DL 3	MINUTE
:T =MIN	
:L DR 3	SECOND
:T =SEC	
:A F 11.1	DISPLAY CLOCK MODE
:= =MODE	MODE = 1, IN 12-HOUR MODE
:BE	

10.5.2 Storing the Updated Time/Date after a RUN to STOP or RUN to POWER OFF Transition

When the S5-95F leaves the RUN mode, it stores the time of terminating program processing in the clock data area in DW18 to DW 21.

The S5-95F stores the time of the last RUN to STOP or RUN to POWER OFF transition always in the clock data area if

- you have programmed saving of the time with COM 95F
  or
- set bit 5 in the status word.

10.5.3 Setting the Prompt Time

After a manual overall reset, the clock data area is assigned the default clock parameters stored in DB1. The following section describes how you can change these parameters in the control program.

Transferring Settings to the Clock

- You can store the settings in the clock data area by using transfer operations (see Table 10-1).

- The AM/PM flag (bit number 7) is only significant in 12-hour mode.
  Bit 7=1 means PM
  Bit 7=0 means AM

- You must transfer the clock data in BCD code.
  The “KC” data format loads a BCD constant into ACCU 1 and is therefore especially suitable.

- If you enter the value “255_D” or “FF_H” in a byte as the prompt time, this byte will be ignored when evaluating “Prompt time reached”. This makes it easy to program, for example, an alarm that is repeated daily by entering the value in the “255_D” or “FF_H”in the “Weekday”, “Date” and “Month” settings.

- You can transfer the prompt time settings to the clock by initiating bit 14 in the status word.

- The settings are transferred 1 second after the start of the next cycle.

- Bit 12 in the status word displays incorrect settings.

Prompt Time Sequence

- Bit 13 in the status word is set after the prompt time has elapsed.
- Bit 13 remains set until you reset it in the control program.
- The prompt time can be read at any time.

> **Caution**
>
> If the prompt time is reached in the STOP mode or during Power OFF, the prompt time cannot be evaluated. It is always deleted on restart.

Example: Setting and evaluating the prompt time

In the example program, the status of input I 0.6 determines whether the settings for the prompt time are transferred. Before setting input I 0.6, you must transfer the settings to flag bytes FY130 and FY135. Enter values that you do not wish to be evaluated as FF_H.

You set the clock mode with input I 1.0. Use input I 0.1 to specify the AM/PM bit for 12-hour mode.

If the preset prompt time has been reached, set flag F 13.2. If errors are made while entering the prompt time, the error bit, flag F 12.2, is set.

The clock data is stored in DB200 beginning with data word DW0, and the status word is data word DW 22.

OB1 STL	Explanation
:	==========
:A I 0.6	SETTING AND EVALUATING THE PROMPT TIME
:AN F 0.3	==========
:S F 20.1	LOAD VALUES INTO FY130 TO FY135
:	FIRST.
:A I 0.6	TRIGGER SETTING OF PROMPT TIME
:= F 0.3	BY SETTING F 20.1 (RESET IN FB11)
:	
:JU FB 11	
NAME :SET PROMPT TIME	
WDAY : FY 130	WEEKDAY
DAY : FY 131	DAY
MON : FY 132	MONTH
HOUR : FY 133	HOUR
AMPM : I 1.1	AMPM-BIT (ONLY IMPORTANT IN 12-HOUR MODE)
MIN : FY 134	MINUTES
SEC : FY 135	SECONDS
ERR : F 12.2	ERROR BIT
ALRM : F 13.2	DISPLAYS THAT PROMPT TIME IS REACHED.
MODE : I 1.0	CLOCK MODE
:BE	

FB11 STL	Explanation
NAME :SET PROMPT TIME	SETTING THE PROMPT TIME
DES :WDAY I/Q/D/B/T/C: I BI/BY/W/D: BY	
DES :DATE I/Q/D/B/T/C: I BI/BY/W/D: BY	
DES :MON I/Q/D/B/T/C: I BI/BY/W/D: BY	
DES :HOUR I/Q/D/B/T/C: I BI/BY/W/D: BY	
DES :AMPM I/Q/D/B/T/C: I BI/BY/W/D: BI	
DES :MIN I/Q/D/B/T/C: I BI/BY/W/D: BY	
DES :SEC I/Q/D/B/T/C: I BI/BY/W/D: BY	
DES :ERR I/Q/D/B/T/C: Q BI/BY/W/D: BI	
DES :ALRM I/Q/D/B/T/C: Q BI/BY/W/D: BI	
DES :MODE I/Q/D/B/T/C: I BI/BY/W/D: BI	
:C DB 200	CALL PARAMETER CONTROL DB
:AN =MODE	24 HOUR MODE = 0, 12 HOUR = 1
:JC =CONT	
:SU D 22.1	CLOCK MODE: STATUS WORD BIT 1
CONT :	SET CLOCK MODE
:TB D 22.13	DISPLAY PROMPT TIME REACHED
:S =ALRM	(BIT 13 IN STATUS WORD)
:TBN D 22.13	
:JC =W002	
:RU D 22.13	RESET BIT AFTER EVALUATION
W002 :	
:A F 20.1	START MONITORING TIME
:L KT 200.0	
:SE T 11	
:AN F 20.1	FLAG IS RESET IF SETTINGS
:JC =M001	HAVE ALREADY BEEN READ INTO THE
:R F 20.1	CLOCK DATA AREA
:	
:	CLOCK DATA AREA
:L =WDAY	STORE VALUE FOR WEEKDAY
:T DR 8	
:L =DAY	STORE VALUE FOR DATE
:T DL 9	
:L =MON	STORE VALUE FOR MONTH
:T DR 9	

FB11 STL (continued)	Explanation
:L =HOUR	STORE VALUE FOR HOURS
:ON =AMPM	IF AM/PM = 1 (AFTERNOON) AND
:ON =MODE	12-HOUR MODE IS SET, THE
:JC =MORN	CORRESPONDING BIT IN THE CLOCK
:L KH 0080	DATA AREA IS SET
:OW	
MORN :T DR 10	
:L =MIN	STORE VALUE FOR MINUTES
:T DL 11	
:L =SEC	STORE VALUE FOR SECONDS
:T DR 11	
:SU D 22.14	TRANSFER SETTINGS
:	(Bit 14 IN STATUS WORD)
M001 :A T 11	BEC, IF MONITORING TIME NOT YET
:BEC	ELAPSED
:TBN D 22.14	HAVE SETTINGS BEEN TRANSFERRED?
:JC =M002	IF YES, JUMP TO M002
:S =ERR	IF ERROR, SET ERROR BIT
:BEU	
M002 :TBN D 22.12	ERROR WHEN ENTERING SETTINGS?
:RB =ERR	IF NO, RESET ERROR BIT
:BEC	BEC, IF NO ERROR
:S =ERR	IF ERROR, SET ERROR BIT
:BE	

10.5.4 Setting the Operating Hours Counter

You can enable the operating hours counter with bit 9 of the status word. This allows you to find out, for example, for how many hours a motor has been in operation. The operating hours counter is active only in the RUN mode.

Transferring Settings to the Operating Hours Counter

You can preset the operating hours counter with a certain start value (e.g. after exchanging the CPU).

- The clock data must be transferred in BCD code.
  The "KC" data format loads a BCD constant into ACCU 1 and is therefore especially suitable for entering the settings.

- If you do not want a value (for example minutes) to be transferred, enter the relevant byte as "255_D" or""FF_H". The current value for this variable is then retained.

- After you have transferred the settings to the clock data area, you must set bit 10 in the status word for the clock to accept the clock data.

- Bit 8 in the status word displays incorrect settings.

Example: Setting the operating hours counter

The status of input I 0.7 determines whether the operating hours counter values are transferred. You must transfer these values to flag bytes FY136 to FY140 before setting input I 0.7 (not implemented in the example program). Values that are not to be changed should be preset with FF_H.

Errors are displayed in flag F 12.3.

The clock data area is in data block DB200 beginning with data word DW22, and the status word is data word DW 22.

OB1 STL	Explanation
: :A I 0.7 :AN F 0.4 :S F 20.2 : :A I 0.7 := F 0.4 : :JU FB 12 NAME :SET OPER. HOURS COUNTER SEC : FY 136 MIN : FY 137 HOUR0: FY 138 HOUR2: FY 139 HOUR4: FY 140 ERR : F 12.3 :BE	========== SETTING THE OPERATING HOURS COUNTER ========== LOAD VALUES INTO FY136 TO FY140 TRIGGER TRANSFER OF SETTINGS FOR OPERATING HOURS COUNTER BY SETTING F 20.2 SECONDS MINUTES HOURS HOURS X 100 ERRORS X 10000 ERROR BIT

FB12 STL	Explanation
NAME :SET OPER. HOURS COUNTER	SETTING THE OPERATING HOURS COUNTER
DES :SEC I/Q/D/B/T/C: I BI/BY/W/D: BY	
DES :MIN I/Q/D/B/T/C: I BI/BY/W/D: BY	
DES :HOUR0 I/Q/D/B/T/C: I BI/BY/W/D: BY	
DES :HOUR2 I/Q/D/B/T/C: I BI/BY/W/D: BY	
DES :HOUR4 I/Q/D/B/T/C: I BI/BY/W/D: BY	
DES :ERR I/Q/D/B/T/C: Q BI/BY/W/D: BI	
:C DB 200	PARAMETER CONTROL DB
:A F 20.2	START MONITORING TIME
:L KT 200.0	
:SE T 12	
:AN F 20.2	FLAG IS RESET IF SETTINGS
:JC =M001	ALREADY READ INTO THE
:R F 20.2	CLOCK DATA AREA
:	
:	CLOCK DATA AREA
:L =SEC	STORE VALUE FOR SECONDS
:T DR 15	
:L =MIN	STORE VALUE FOR MINUTES
:T DL 16	
:L =HOUR0	STORE VALUE FOR HOURS
:T DR 16	
:L =HOUR2	STORE VALUE FOR HOURS X 100
:T DL 17	
:L =HOUR4	STORE VALUE FOR HOURS X 10,000
:T DR 17	
:SU D 22.10	TRANSFER SETTINGS
:	(BIT 10 IN STATUS WORD)
:SU D 22.9	ENABLE OPERATING HOURS COUNTER
:	(BIT 9 IN STATUS WORD)
M001 :A T 12	BEC IF MONITORING TIME NOT YET
:BEC	ELAPSED
:TBN D 22.10	HAVE SETTINGS BEEN TRANSFERRED?
:JC =M002	IF YES, JUMP TO M002
:S =ERR	IF ERROR, SET ERROR BIT
:BEU	
M002 :TBN D 22.8	ERROR WHEN ENTERING SETTINGS?
:RB =ERR	IF NO, RESET ERROR BIT
:BEC	BEC IF NO ERROR
:S =ERR	IF ERROR, SET ERROR BIT
:BE	

Reading the Current Operating Hours Counter

The current data is stored in words 12 to 14 of the clock data area. You can use load operations to read out the data.

Example: Reading the operating hours counter

You need to switch off a machine for inspection after every 300 hours of operation. Flag F 12.4 is set when the machine is switched off. After 300 hours of operation, a jump is made to PB5 to switch the machine off (not programmed in the example).

The clock data area is in DB200 beginning with data word DW0, and the status word is data word DW22.

OB1 STL	Explanation
:JU FB 14 NAME :BETR-LES : :BE	EVALUATE OPERATING HOURS COUNTER

FB14 STL	Explanation
NAME :BETR-LES	READING THE OPERATING HOURS COUNTER
:C DB 200	DB IN WHICH THE CLOCK DATA IS LOCATED.
:A F 12.4	IF AUXILIARY FLAG 12.4 IS SET,
:BEC	THE MACHINE IS ALREADY OFF.
:	BLOCK END
:L DL 14	LOAD HOUR VALUE X 100 IN ACCU 1
:	
:L KC 003	COMPARE TO 3 (=300 HOURS)
:><F	
:BEC	END IF 300 HOURS NOT YET
:	REACHED
:S F 12.4	SET AUXILIARY FLAG
:JU PB 5	JUMP TO PB5 WHEN 300 OPERATING
:	HOURS REACHED
:	
:BE	

11 Analog Value Processing

11.1 Analog Input Modules (Type P) 11- 1

11.2 Connecting Current and Voltage Sensors to Analog Input Modules 11- 1
11.2.1 Voltage Measuring with Isolated/Non-Isolated Thermocouples 11- 2
11.2.2 Two-Wire Connection of Voltage Sensors 11- 3
11.2.3 Two-Wire Connection of Current Sensors 11- 4
11.2.4 Connection of Two-Wire and Four-Wire Transducers 11- 5

11.3 Start-Up of Analog Input Modules 11- 7

11.4 Analog Value Representation of Analog Input Modules 11- 8

11.5 Analog Output Modules (Type W) 11- 10
11.5.1 Connection of Loads to Analog Output Modules 11- 10
11.5.2 Analog Value Representation of Analog Output Modules 11- 11

11.6 Analog Value Conversion: Function Blocks FB250 and FB251 . . . . 11- 13
11.6.1 Reading in and Scaling an Analog Value - FB250 - 11- 13
11.6.2 Output of Analog Value - FB251 - 11- 16

11.7 Failsafe Analog Value Processing - FB232 and FB233 - 11- 18
11.7.1 Discrepancy Analysis for Two Analog Inputs - FB232 - 11- 19
11.7.2 Parameterization Example for FB232 . . . . . 11- 27
11.7.3 Discrepancy Analysis for Three Analog Inputs - FB233 - 11- 28
11.7.4 Combining FB232 and FB233 . . . . . 11- 39

11.8 Circuit Versions for Function Blocks FB232 and FB233 . . . . . 11- 39
11.8.1 Circuit Type R4.2 for AK4 . . . . . 11- 42
11.8.2 Circuit Type R4.4 for AK4 . . . . . 11- 43
11.8.3 Circuit Type R5.1 for AK5 . . . . . 11- 44
11.8.4 Circuit Type R6.1 for AK6 . . . . . 11- 45
11.8.5 Circuit Type R6.2 for AK6 . . . . . 11- 47
11.8.6 Circuit Type R6.3 for AK6 . . . . . 11- 49
11.8.7 Circuit Type R6.4 for AK6 . . . . . 11- 51
11.8.8 Circuit Type R6.5 for AK6 . . . . . 11- 53

Figures

11-1 Voltage Measuring with Isolated Thermocouples (6ES5 464-8MA21) 11- 2
11-2 Voltage Measuring with Non-Isolated Thermocouples (6ES5 464-8MA21) . . 11- 2
11-3 Two-Wire Connection of Voltage Sensors (6ES5 464 8MB11 and 6ES5 464-8MC11) 11- 3
11-4 Two-Wire Connection for Current Sensors (6ES5 464-8MD11) 11- 4
11-5 Connection of Two-Wire Transducers (6ES5 464-8ME11) 11- 5
11-6 Connection for Four-Wire Transducers (6ES5 464-8ME11) 11- 6
11-7 Load Connection via a Four-Wire Circuit (6ES5 470-8MA12) 11- 10
11-8 Load Connection via a Two-Wire Circuit (6ES5 470-8MB12) 11- 11
11-9 Scaling Schematic for FB250 . . . . . 11- 13
11-10 Schematic for “Display of Tank Make-Up Quantity” 11- 14
11-11 Conversion of the Nominal Range into the Defined Range 11- 14
11-12 Schematic for “Display of Tank Contents” 11- 16
11-13 Transformation of the Analog Value to the Nominal Range 11- 17
11-14 Scaling Schematic for a Scaling Range of -1000 to +1000 . . . . . 11- 23
11-15 Diagnostic Byte in DR 4 . . . . . 11- 26
11-16 Discrepancy Test for the Three Measured Values 11- 33
11-17 Scaling Schematic for a Scaling Range of -1000 to +1000 . . . . . 11- 34
11-18 Diagnostic Byte in DW15 . . . . . 11- 37

Tables

11-1 Operating Mode Switch Settings for Analog Input Modules 464-8M.11 . . . . 11- 7
11-2 Representation of an Analog Input Value as a Bit Pattern 11- 8
11-3 Analog Input Module 464-8MC11, -8MD11 (Bipolar Fixed-Point Number) 11- 8
11-4 Analog Input Module 464-8ME11, 4×4 to 20 mA (Absolute Value Representation) 11- 9
11-5 Representation of an Analog Output Value as a Bit Pattern 11- 11
11-6 Output Voltages and Currents for Analog Output Modules (Fixed-Point Number Bipolar) 11- 12
11-7 Output Voltages and Currents for Analog Output Modules (Unipolar) 11- 12
11-8 Call and Parameter Assignments of FB250 . . . . . 11- 13
11-9 Call and Parameter Assignments of FB251 . . . . . 11- 16
11-10 FB232 Block Parameters 11- 20
11-11 Operands Reserved by FB232 . . . . . 11- 22
11-12 FB233 Block Parameters 11- 29
11-13 Reserved Operands 11- 31
11-14 Qualilty Levels for Failsafe Analog Input Modules 11- 40
11-15 Intervals for the AI 464-8MG11 Function Test at the User Level 11- 41
11-16 Circuit Diagram for Type R4.2 for AK4 . . . . . 11- 42
11-17 Circuit Diagram for Type R4.4 for AK4 . . . . . 11- 43
11-18 Circuit Diagram for Type R5.1 for AK5 . . . . . 11- 44
11-19 Circuit Diagram for Type R6.1 for AK6 . . . . . 11- 45
11-20 Circuit Diagram for Type R6.2 for AK6 . . . . . 11- 47
11-21 Circuit Diagram for Type R6.3 for AK6 . . . . . 11- 49
11-22 Circuit Diagram for Type R6.4 for AK6 . . . . . 11- 51
11-23 Circuit Diagram for Type R6.5 for AK6 . . . . . 11- 53

11 Analog Value Processing

Failsafe function blocks for analog value conversion are integrated in the S5-95F. In conjunction with the 464-8MG11 analog module, these function blocks enable failsafe analog value processing.

For non-failsafe analog value processing, the S5-95F contains reaction-free function blocks which can be used together with numerous modules from the S5-100U product series. The analog modules which can be used for this purpose are listed in Appendix A.

The following sections describe:

- Connection of sensors and actuators
- Analog value representation of modules
- Settings that you must carry out on the modules
- Examples for using the integrated analog value conversion blocks FB232 and FB232 für die sicherheitsgerichtete Analogwertverarbeitung
- Examples for using the integrated analog value conversion blocks FB250 and FB 251 für die nichtsicherheitsgerichtete Analogwertverarbeitung.

11.1 Analog Input Modules (Type P)

Analog input modules convert analog process signals to digital values that the CPU can process (via the process image input table, PII). In the following sections, you will find information about the operating principle, wiring methods, and start-up and programming of analog input modules.

11.2 Connecting Current and Voltage Sensors to Analog Input Modules

Observe the following rules to connect current and voltage sensors to analog input modules:

- When you have multi-channel operations, assign the channels in ascending order. This shortens the data cycle.

- Use terminals 1 and 2
  - for the connection of a compensating box (464-8MA11)
    or
  - for the supply of two-wire transducers (464-8ME11).

  Terminals 1 and 2 cannot be used with the remaining analog input modules.

- Short-circuit the terminals of unused inputs.

- Set the reference potentials of the sensors to a common reference potential. Do this to prevent the potential difference between the common references from exceeding 1 V.

11.2.1 Voltage Measuring with Isolated/Non-Isolated Thermocouples

The **464-8MA21** module is suitable for voltage measuring with thermocouples. In the case of **isolated sensors**, such as isolated thermocouples, the permissible potential difference V_{CM} between the inputs' negative terminals and the standard mounting rail's potential may not be exceeded. To prevent this, the sensor's negative potential must be connected to the central earthing point (Figure 11.1).
If no compensating box is used, terminals 1 and 2 must be short-circuited!

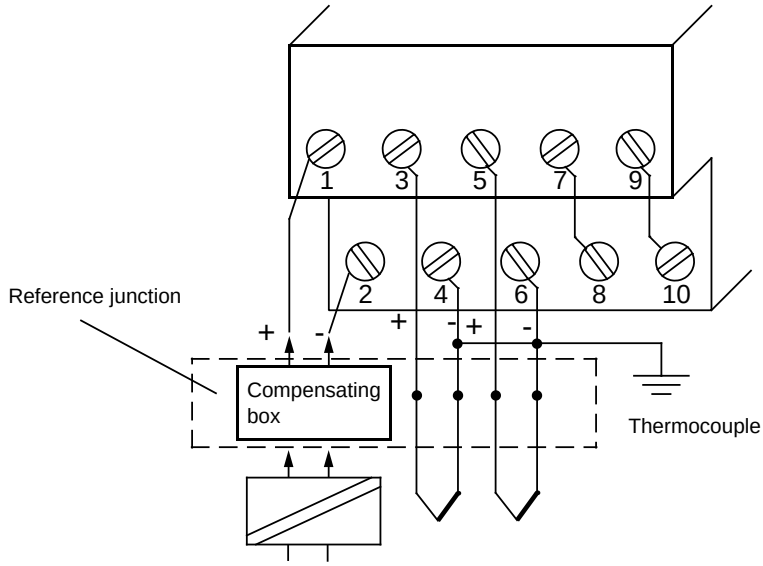


Figure 11-1. Voltage Measuring with Isolated Thermocouples (6ES5 464-8MA21)

In the case of **non-isolated** sensors, for example non-isolated thermocouples, the maximum permissible potential difference may not exceed V_{CM}.

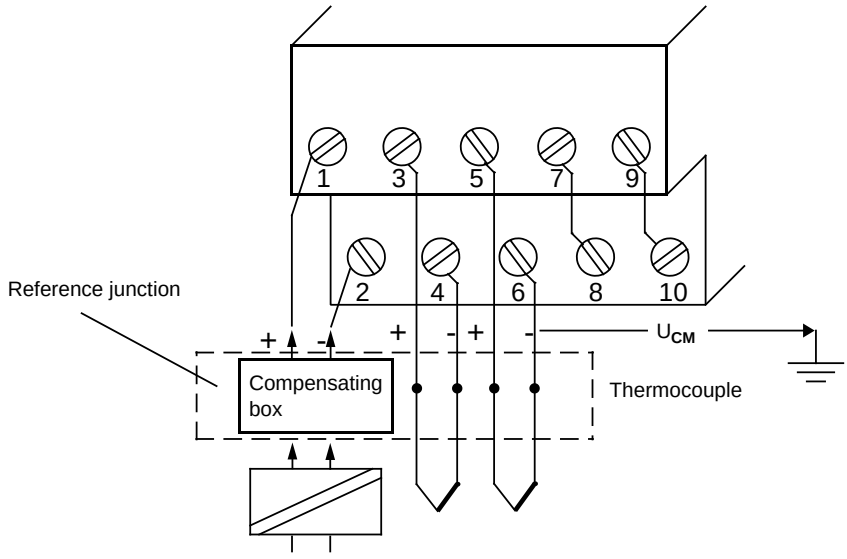


Figure 11-2. Voltage Measuring with Non-Isolated Thermocouples (6ES5 464-8MA21)

Connecting Thermocouples with Compensating Box to 464-8MA21 Module

The influence of the temperature on the reference junction (e.g. terminal box) can be made up for by using a compensating box.

Note the following:

- The compensating box must be voltageless.
- The power supply unit must have a grounded shield winding.
- The compensation box must be connected to terminals 1 and 2 of the terminal block.

11.2.2 Two-Wire Connection of Voltage Sensors

The following analog input modules are available for connecting voltage sensors:

- **464-8MB11** for voltages of ±1 V
- **464-8MC11** for voltages of ±10 V

Wire as shown in Figure 11.3.

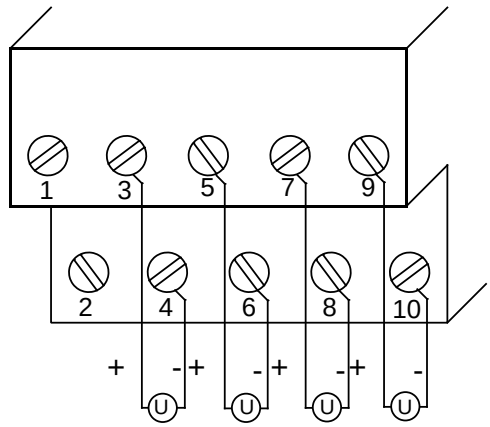


Figure 11-3. Two-Wire Connection of Voltage Sensors (6ES5 464-8MB11 and 6ES5 464-8MC11)

11.2.3 Two-Wire Connection of Current Sensors

You can use module **464-8MD11** for the two-wire connection of current sensors.

Wire as shown in Figure 11.4.

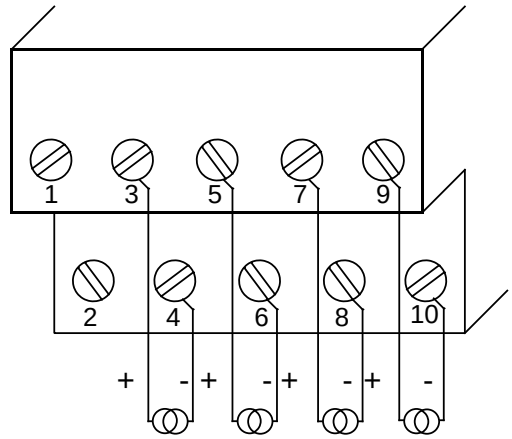


Figure 11-4. Two-Wire Connection for Current Sensors (6ES5 464-8MD11)

11.2.4 Connection of Two-Wire and Four-Wire Transducers

The **464-8ME11** analog input module provides 24 V inputs 1 and 2 to supply two-wire transducers. The two-wire transducer then converts the supplied voltage into a current of 4 to 20 mA.

Wire as shown in Figure 11.5.

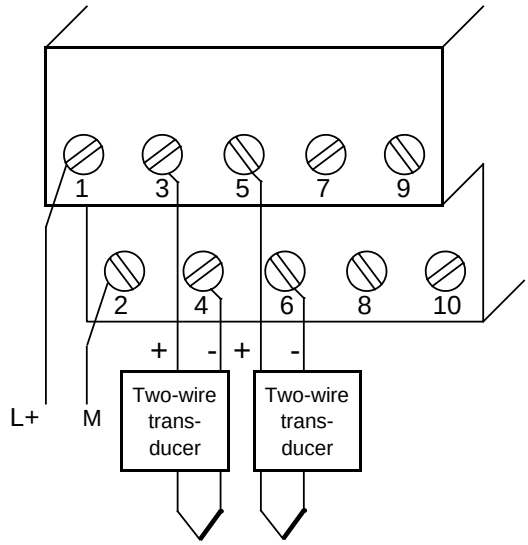


Figure 11-5. Connection of Two-Wire Transducers (6ES5 464-8ME11)

If you use a four-wire transducer connect it as shown in Figure 11-6.

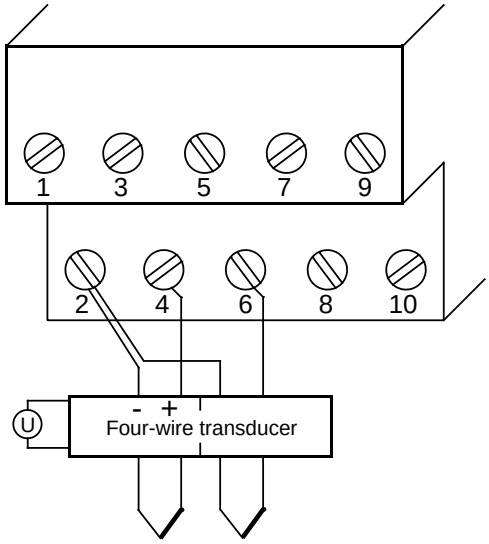


Figure 11-6. Connection for Four-Wire Transducers (6ES5 464-8ME11)

Four-wire transducers require their own power supply. Connect the "+" pole of the four-wire transducer to the corresponding "-" pole of the terminal block (a connection technique that is the opposite of the two-wire transducer).

Connect negative terminals of the four-wire transducer to terminal two of the terminal block.

Inputs 4, 6, 8, and 10 of the analog input module **464-8ME11** are connected internally via shunt resistors. Because of the internal shunt resistors, broken wire signalling is not possible.

11.3 Start-Up of Analog Input Modules

Set the intended operating mode using the switches on the front panel of **464-8M.11** analog input modules. These switches are located on the right side at the top of the front panel of the module.

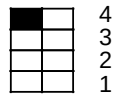
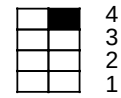
Power supply frequency: Set the switch to the available power supply frequency. This selects the integration time of the A/D converters for optimal interference voltage suppression.
Power frequency 50 Hz: Integration time 20 ms
Power frequency 60 Hz: Integration time 16.66 ms

Operation: Set the number of channels you wish to assign on the input module. If there are fewer than four channels, less address space will be assigned and measured values will be updated faster.

Broken wire: Once the broken wire signal has been activated, a break on one of the lines to the sensor (thermocouple or PT 100) or of the sensor itself causes the red LED above the function selection switch to light up. At the same time, the broken wire error bit F (bit 1, byte 1) for the faulty channel is set.

The module " recognizes" a wire break by applying a conventional tripping current to the input terminals and by comparing the resulting voltage to a limit value. If there is a wire break in the sensor or the lines, the voltage exceeds the limit value and a "wire break" signal is sent. When the signal at the input is measured with a digital voltmeter, the tripping current pulses cause apparent fluctuations of the signal. Deactivation of the wire break signal does **not** turn off the tripping current.

Table 11-1. Operating Mode Switch Settings for Analog Input Modules 464-8M.11

Function	Settings for Operating Mode Switch		
Power supply frequency	50 Hz 	60 Hz 	
Operation	1 channel (channel 0) 	2 channels (channel 0 and channel 1) 	4 channels (channel 0 to channel 3) 
Wire break	With wire break signal 	No wire break signal 	

11.4 Analog Value Representation of Analog Input Modules

Each analog process signal has to be converted into a digital format, to be stored in the process image input table (PII). The analog signals are converted into a binary digit that is written in two bytes:

Analog values are represented in two's complement.
Each bit position has a fixed value in powers of two (see Tables 11-2 and 11-5)

The following tables show the analog value representations of the different analog inputs in 2-byte format. You will need this information to program FB250 and FB251 (see section 11.6).

Table 11-2. Representation of an Analog Input Value as a Bit Pattern

	High Byte								Low Byte							
Bit Number	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
Analog Value Represent.	S	2^{11}	2^{10}	2^{9}	2^{8}	2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}	X	E	OV

Key:

S	Sign bit	0="+", 1="-"
X	Irrelevant bits	
E	Error bit	0= no wire break; 1=wire break
OV	Overflow bit	0= Measured value 4095 units at the most 1= Measured value greater than or equal to 4096 units

Table 11-3. Analog Input Module 464-8MC11, -8MD11 (Bipolar Fixed-Point Number)

Units	Measured Value in V	Measured Value in mA	High Byte								Low Byte								Range
>4095	20.000	40.0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	1	Overflow
4095	19.995	39.9902	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	Overrange
2049	10.0048	20.0098	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	
2048	10.000	20.0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
1024	5.000	10.0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
1	0.0048	0.0098	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	
0	0.0	0.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Nominal range
-1	-0.0048	-0.0098	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	
-1024	- 5.000	-10.0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
-2048	-10.000	-20.0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
-2049	-10.0048	-20.0098	1	0	1	1	1	1	1	1	1	1	1	1	1	0	0	0	Overrange
-4095	-19.995	- 39.9902	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	
<-4095	-20.000	-40.0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	Overflow

464-8MC11 (4x±10 V)
464-8MD11 (4x±20 mA)

Table 11-4. Analog Input Module 464-8ME11, 4x4 to 20 mA (Absolute Value Representation)

Units	Measured Value in mA	High Byte								Low Byte								Range*
>4095	> 32.769	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	1	Overflow
4095	31.992	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	Overrange
2561	20.008	0	1	0	1	0	0	0	0	0	0	0	0	1	0	0	0	
2560	20.0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	
2048	16.0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Nominal range
512	4.0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	
511	3.992	0	0	0	0	1	1	1	1	1	1	1	1	1	0	0	0	
384	3.0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	
0	0.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Transducer
-1	-0.008	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	failure?
<-4095	<- 32.769	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	Overflow

* Because of tolerances of components used in the module, the converted value can also be negative (e.g. $FFF8_H$ Unit: -1).

11.5 Analog Output Modules (Type W)

Analog output modules convert the bit patterns that are output by the CPU into analog output voltages or currents.

11.5.1 Connection of Loads to Analog Output Modules

No adjustments are necessary if you want to connect loads to the analog outputs.

Check the following items before connecting loads:

- The load voltage 24 V DC must be connected to terminals 1 and 2.
- The maximum permissible potential difference between the outputs is 60 V AC.
- Unused outputs must be left open-circuited.

Figure 11-7 shows how to connect loads to the voltage outputs of the module 470-8MA12 (2x ±10 V):

The sensor lines (S+ and S-) must be directly connected to the load, so that the voltage is measured and regulated directly at the load. In this manner, voltage drops of up to 3 V per line can be compensated for.

The sensor lines can be left out if the resistances of the QV and M lines are negligible compared to the load resistance.

In such a case, connect terminal S+to terminal QV, and terminal S- to M_{ANA}.

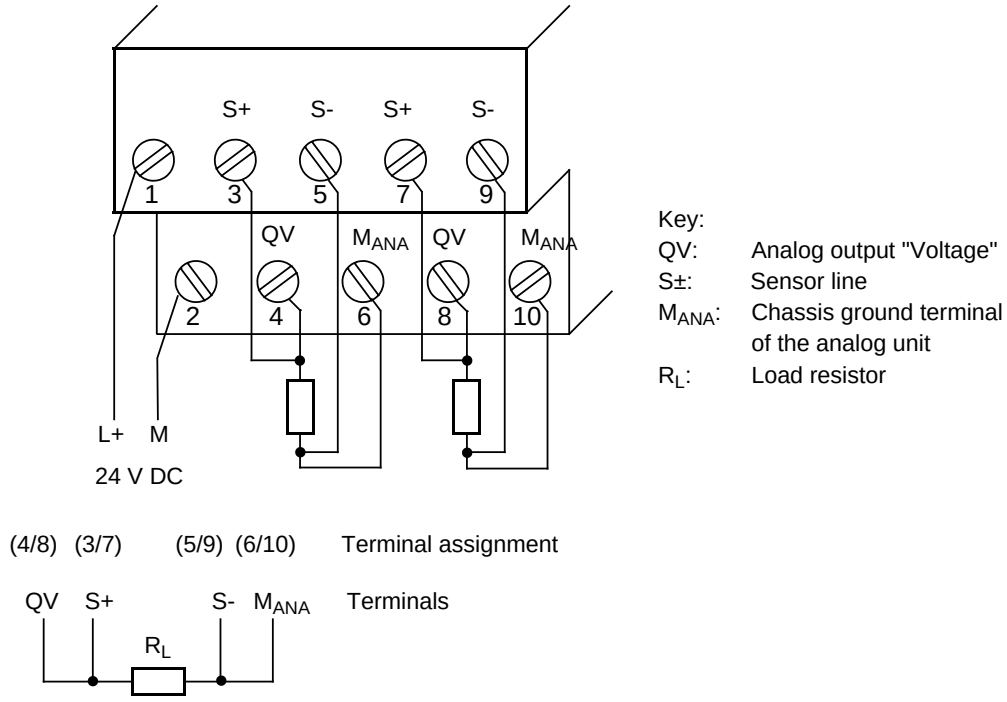


Figure 11-7. Load Connection via a Four-Wire Circuit (6ES5 470-8MA12)

Figure 11-8 shows how to connect loads to the current outputs of the module 470-8MB12 (2 x ±20 mA).

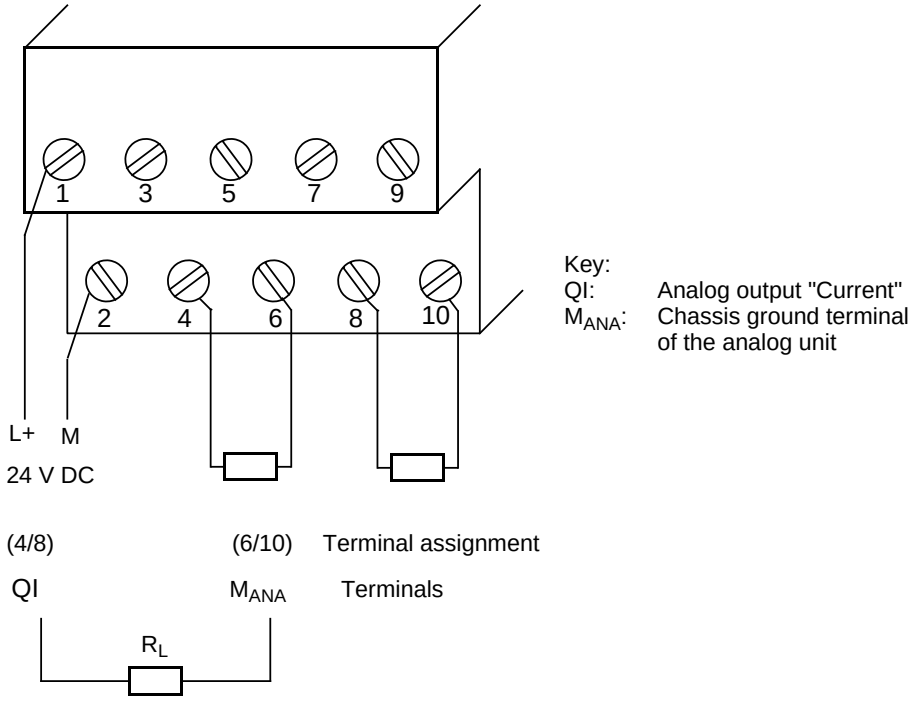


Figure 11-8. Load Connection via a Two-Wire Circuit (6ES5 470-8MB12)

11.5.2 Analog Value Representation of Analog Output Modules

Table 11-5 shows how the analog output value has to be stored in the process image output table (PIQ).

Table 11-5. Representation of an Analog Output Value as a Bit Pattern

	High Byte								Low Byte							
Bit number	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
Analog value represent.	S	2^{10}	2^{9}	2^{8}	2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}	X	X	X	X

Key: X irrelevant bits

Tables 11-6 and 11-7 show the voltage and currents assigned to the bit patterns.

Table 11-6. Output Voltages and Currents for Analog Output Modules (Fixed-Point Number Bipolar)

Units	Output Values in V	Output Values in mA	High Byte	Low Byte	Range
1280	12.5	25.0	0 1 0 1 0 0 0 0	0 0 0 0 x x x x	Overrange
1025	10.0098	20.0195	0 1 0 0 0 0 0 0	0 0 0 0 x x x x	
1024	10.0	20.0	0 1 0 0 0 0 0 0	0 0 0 0 x x x x	
512	5.0	10.0	0 0 1 0 0 0 0 0	0 0 0 0 x x x x	
1	0.0098	0.0195	0 0 0 0 0 0 0 0	0 0 0 0 x x x x	
0	0.0	0.0	0 0 0 0 0 0 0 0	0 0 0 0 x x x x	
-1	-0.0098	-0.0195	1 1 1 1 1 1 1 1	1 1 1 1 x x x x	Nominal range
-512	-5.0	-10.0	1 1 1 0 0 0 0 0	0 0 0 0 x x x x	
-1024	-10.0	-20.0	1 1 0 0 0 0 0 0	0 0 0 0 x x x x	
-1025	-10.0098	-20.0195	1 0 1 1 1 1 1 1	1 1 1 1 x x x x	
-1280	-12.5	-25.0	1 0 1 1 0 0 0 0	0 0 0 0 x x x x	Overrange

2x±10 V 6ES5 470-8MA12
2x±20 mA 6ES5 470-8MB12

Table 11-7. Output Voltages and Currents for Analog Output Modules (Unipolar)

Units	Output Values in V	Output Values in mA	High Byte	Low Byte	Range
1280	---	24.0	0 1 0 1 0 0 0 0	0 0 0 0 x x x x	Overflow
1025	---	20.016	0 1 0 0 0 0 0 0	0 0 0 1 x x x x	
1024	---	20.0	0 1 0 0 0 0 0 0	0 0 0 0 x x x x	
512	---	12.0	0 0 1 0 0 0 0 0	0 0 0 0 x x x x	Nominal range
1	---	4.016	0 0 0 0 0 0 0 0	0 0 0 1 x x x x	
0	---	4.0	0 0 0 0 0 0 0 0	0 0 0 0 x x x x	
-1	---	3.984	1 1 1 1 1 1 1 1	1 1 1 1 x x x x	
-256	---	0.0	1 1 1 1 0 0 0 0	0 0 0 0 x x x x	
-512	---	-4.0	1 1 0 0 0 0 0 0	0 0 0 0 x x x x	Overrange
-1024	---	-12.0	1 1 0 0 0 0 0 0	0 0 0 0 x x x x	
-1280	---	-16.0	1 0 1 1 0 0 0 0	0 0 0 0 x x x x	

2x 4 to 20 mA 6ES5 470-8MC11

11.6 Analog Value Conversion: Function Blocks FB250 and FB251

11.6.1 Reading in and Scaling an Analog Value - FB250 -

Function block FB250 reads in an analog value from an analog input module and outputs a value XA in the scale range specified by the user.

Specify the type of analog value representation for the module (channel type) in the KNKT parameter.

Define the desired range using the "upper limit" (OGR) and "lower limit" (UGR) parameters.

Table 11-8. Call and Parameter Assignments of FB250

Parameter	Explanation	Type	Assignment	STL
BG	Slot number	D KF	0 to 7	: JU FB 250 NAME : RLG:AI BG : KNKT : OGR : UGR : EINZ : XA : FB : BU :
KNKT	Channel number Channel type	D KY	KY = x,y x = 0 to 3 y = 3 to 6 3: Absolute value representation (4 to 20 mA) 4: Unipolar representation 5: Bipolar absolute value 6: Bipolar fixed-point number	
OGR	Upper limit of the output value	D KF	-32767 to +32767	
UGR	Lower limit of the output value	D KF	-32767 to +32767	
EINZ	Not relevant			
XA	Output value	Q W	Scaled analog value is “0” on wirebreak	
FB	Error bit	Q BI	“1” on wirebreak, illegal channel or slot number or illegal channel type	
BU	Range violation	Q BI	“1” when nominal range is exceeded	

Scaling Schematic

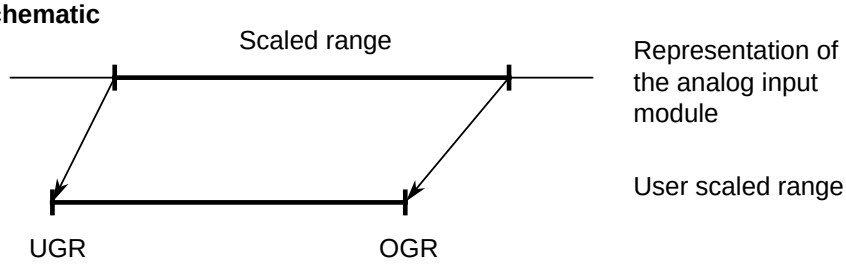


Figure 11-9. Scaling Schematic for FB250

Example: Display of Tank Make-Up Quantity

The make-up of a cylindrical tank holding 30 m^3 is to be shown on a 3-digit display. The individual digits must be set in BCD.

The level of the liquid in the tank is sensed by a SONAR-BERO®, range 80 to 600 cm, with analog output (see Catalog NS3).

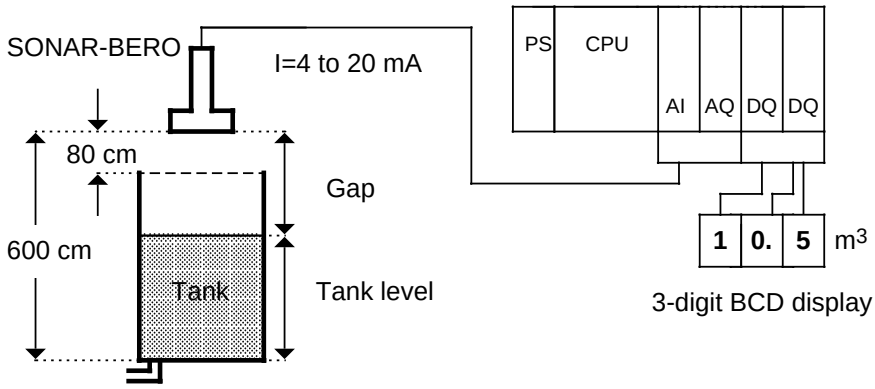


Figure 11-10. Schematic for "Display of Tank Make-Up Quantity"

The analog output of the SONAR-BERO delivers a constant current in the range 4 to 20 mA proportional to the gap between sensor and liquid. This current is routed to the 4 to 20 mA analog input module in slot 0, channel 0.

FB250 converts the range 4 to 20 mA to the range 0 to 30.0 m^3.
The value is stored in flag word 1 as a fixed-point number. Initialization takes place in the calling block. FB241converts the fixed-point number into a BCD number (see FB241).

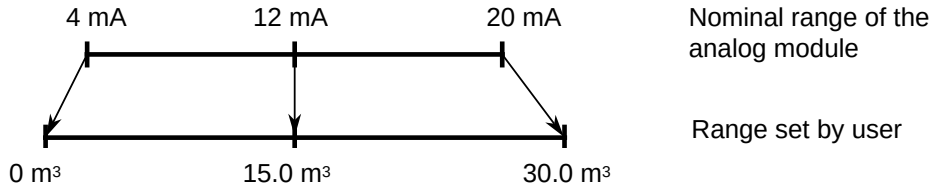


Figure 11-11. Conversion of the Nominal Range into the Defined Range

STL	Explanation
JU FB 250	Unconditional call FB250
NAME : RLG:AI	
BG : 0	Slot 0
KNKT : 0.3	Channel 0, channel type 3
OGR : 300	Upper limit: 30.0 m^3
UGR : 0	Lower limit: 0.0 m^3
EINZ : FY0	No meaning
XA : FY2	Make-up quantity stored in flag word 1 as fixed-point number
FB :F0.0	"1", if wire break
BU :F0.1	"1", if tank too full
JU FB 241	Conversion of fixed-point number into BCD number
. . .	

The BCD number is stored in flag bytes 11 to 13. Output is via two 8-channel digital output modules in slots 4 and 6. The BCD tetrads 5 and 6 stored in flag word 11 need not be output since the number has only three digits.

STL	Explanation
. . .	
L FW12	Read tetrads 0 to 3 of the BCD number
T QW2	and transfer to output modules.
BE	

11.6.2 Output of Analog Value - FB251 -

Analog values can be output to analog output modules using this function block. In doing so, values from the range between the lower limit (UGR) and high limit (OGR) parameters are converted to the nominal range of the module in question.

Table 11-9. Call and Parameter Assignments of FB251

Parameter	Explanation	Type	Assignment	
XE	Analog value to be output	I W	Input value (two's complement) in the range UGR to OGR	: JU FB 251 NAME : RLG:AQ XE : BG : KNKT : OGR : UGR : FEH : BU :
BG	Slot address	D KF	0 to 7	
KNKT	Channel number Channel type	D KY	KY = x,y x = 0;1 y = 0;1 0: unipolar repre-sentation 1: bipolar fixed-point number	
OGR	Upper limit of the output value	D KF	-32767 to +32767	
UGR	Lower limit of the output value	D KF	-32767 to +32767	
FEH	Error in limit value setting	Q BI	"1" if UGR = OGR, invalid channel or slot, or invalid channel type	
BU	Input value exceeds UGR or OGR	Q BI	"1" if XE lies outside limits (UGR; OGR). XE assumes the limit value	

Example: Display of Tank Contents on an Analog Measuring Instrument

The make-up quantity of a 30 m^3 tank is stored in flag word 1 as a fixed-point number (see example FB250). The ± 20 mA analog output module in slot 2, channel 0, transfers the standardized value to the measuring instrument. The value is displayed within the range 0 to 20 mA.

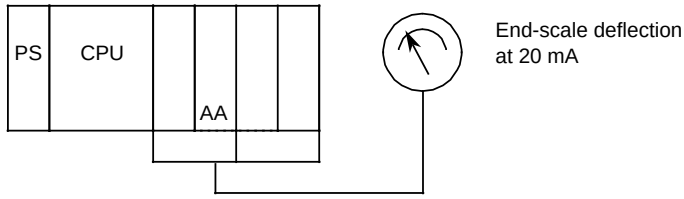


Figure 11-12. Schematic for "Display of Tank Contents"

The tank contents are determined from the make-up quantity.

STL	Explanation
L KF +300	Maximum tank capacity
L FW 1	Make-up quantity
-F	Calculate difference
T FW 20	Store tank contents in FW20
...	

The UGR and OGR parameters of FB 251 refer to the nominal range of the analog output module. For this reason, the UGR parameter must be assigned the value -300.

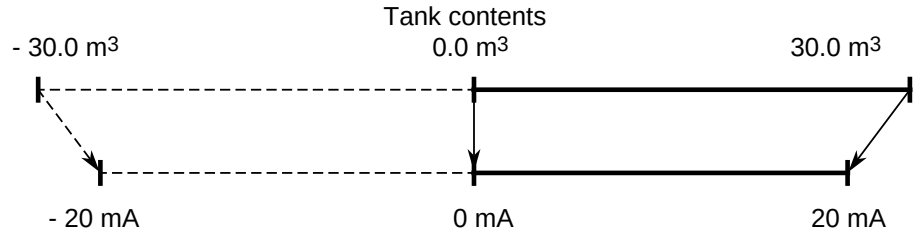


Figure 11-13. Transformation of the Analog Value to the Nominal Range

STL	Explanation
...	
JU FB251	Unconditional call-up FB251
NAME :RLG:AQ	
XE :FW20	Tank contents
BG :2	Slot 2
KNKT :0.1	Channel 0, channel type 1
OGR :300	Upper limit 30.0 m3
UGR :-300	Lower limit - 30.0 m3
FEH :F0.2	“1”, if wire break
BU :F0.3	“1”, if tank too full
BE	

11.7 Failsafe Analog Value Processing -FB 232 and FB 233-

Failsafe analog value processing can be implemented in an S5-95F system in conjunction with at least two 6ES5 464-8MG11 analog modules and function block FB 232 or FB 233.

Prerequisites:

- If you use **non-failsafe** analog sensors, those sensors must at least provide **diversity** and **redundancy**;
- The analog sensors must be supplied by the same **load voltage** source as the CPUs so that the analog sensors can profit from the high quality of the CPUs' voltage monitors. The sensors must function without error within the specified voltage limits.
- If you work with **static** analog sensor values, you will, as a rule, have more circuitry overhead than is the case for dynamic analog sensor values (see circuit versions in section 11.8).
  An analog sensor value is **dynamic** when it goes through the operational value range at least once within 24 hours.

The availability of the analog input modules can also be increased by using redundant I/Os.

Blocks FB 232 and FB 233 monitor the assigned analog inputs for permissible deviations and combine the analog inputs into a standardized output signal. Function blocks FB 232 and FB 233 can be used separately or in combination. They monitor the analog signals and set the standardized signal to a safe value when deviations are detected.

Depending on the circuit version, the blocks can be used for quality levels AK4, AK5 and AK6.

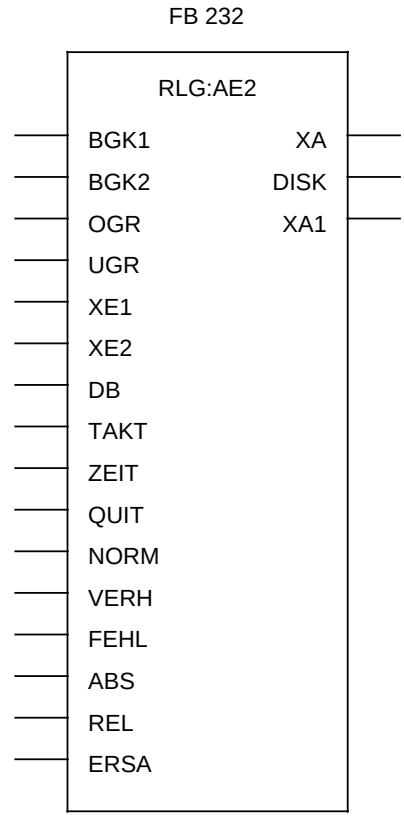
11.7.1 Discrepancy Analysis for Two Analog Inputs -FB 232-

FB 232 Block Parameters

STL

```
0000        :JU FB     232
0001  NAME   :RLG:AE2
0002  BGK1   :
0003  BGK2   :
0004  OGR    :
0005  UGR    :
0006  XE1    :
0007  XE2    :
0008  DB     :
0009  TAKT   :
000A  ZEIT   :
000B  QUIT   :
000C  NORM   :
000D  VERH   :
000E  FEHL   :
000F  ABS    :
0010  REL    :
0011  ERSA   :
0012  XA     :
0013  DISK   :
0014  XA1    :
0015        :
0016        :BE
```

CSF



Note

The data block specified in the input parameter DB is specifically allocated to the two analog channels read (parameters BGK1 and BGK2 or XE1 and XE2).
If you read two additional analog channels with FB 232, you must specify another data block.

Table 11-10. FB 232 Block Parameters

Parameter	Description	Type		Contents	Remarks
BGK1	Module slot no./channel no. of the 1st analog input	D	KY	KY = x,y x = 0 to 7 (slot) y = 0 to 3 (channel) Exception: KY = 255,255 when using direct block input XE1	
BGK2	Module slot no./channel no. of the 2nd analog input	D	KY	KY = x,y x = 0 to 7 (slot) y = 0 to 3 (channel) Exception: KY = 255,255 when using direct block input XE2	
OGR	Upper limit of the scaled range	D	KF	-8192 to +8191	
UGR	Lower limit of the scaled range	D	KF	-8192 to +8191	
XE1	Direct block input for a non-scaled analog value in place of 1st analog channel BGK1	I	W	FW 0 to FW198 DW 6 to DW 255 of the parameter control OB Value range: -512 to +3583	*1), *2)
XE2	Direct block input for a non-scaled analog value in place of 2nd analog input BGK2	I	W	Same as parameter XE1	*1), *2)
DB	Auxiliary data block for storing old values, error states and user data.	B		DB 2 to DB 251 Minimum length 6 DW	
TAKT	Clock pulse for time-restricted availability	I	BI	I/O/F bit e.g. clocked in seconds	
ZEIT	Time in CLOCK units for time-restricted availability	D	KF	0 to +32767	

Table 11-10. FB 232 Block Parameters (continued)

Parameter	Description	Type		Contents	Remarks
QUIT	Acknowledgement input for error acknowledgement	I	BI	I/Q/F bit	
NORM	Preselection of generation of the output signal in the event of error-free operation	D	KC	MW = Average value MA = Maximum value MI = Minimum value	
VERH	Preselection of the error response to a discrepancy error	D	KC	SA = Immediate shutdown ZV = Time-restricted availability	
FEHL	Preselection of generation of the output signal in the event of a discrepancy error	D	KC	ER = Substitute value MA = Maximum value MI = Minimum value	
ABS	Permissible absolute error tolerance based on the as yet unscaled input value	D	KF	0 to +2048 0 = Disconnected	
REL	Permissible relative error tolerance based on the non-scaled input value	D	KF	0 = Disconnected 1 = 3.125% 2 = 6.25% 3 = 12.5% 4 = 25% 5 = 50%	
ERSA	Substitute value as non-scaled input value	D	KF	-512 to +3584	
XA	Block output for standardized and scaled measured value	Q	W	FW 0 to FW198 DW6 to DW255 QW	*1)
DISK	Output bit, discrepancy error (acceptance value = 1)	Q	BI	I/Q/F bit	
XA1	Output for standardized measured value without scaling	Q	W	FW 0 to FW198 DW6 to DW255 QW	*1)

*1) If data words are used for these parameters, they must lie in the data block specified in the DB parameter. Data words 0 to 5 are used internally by the function block, and may not be used for this purpose. If the parameter is not needed, a free data word in the initialized data block should also be specified here.

*2) A discrepancy is flagged when the permissible value range is exceeded.

Execution Times and Operands Reserved by FB 232

FB 232's execution time is approximately 5 ms. The function block reserves the operands listed below.

Table 11-11. Operands Reserved by FB232

Type	Reserved Area
Temporary flag	FY 240 to FY 255
Data words	DW 0 to DW 5 in the assigned DB

FB 232 Functionality

The block can perform the functions listed below, which will be discussed in more detail in the following.

- Reading of two analog signals
- Discrepancy analysis
- Signal standardization
- Scaling of the output signal
- Error response to discrepancy errors

Reading the Analog Signals

As standard function, the block reads two analog channels.
Specification of the module slot and the channel number, both of which are part of parameters BGK1 and BGK2, suffices to define the address of the analog signals to be read.

Discrepancy Analysis

A discrepancy analysis is carried out for both of the analog signals that were read.
This includes checking for a maximum permissible ***absolute*** discrepancy ***and*** a maximum permissible ***relative*** discrepancy.
A discrepancy error is detected when the discrepancy between the two analog values has exceeded both the absolute and the relative limit value.

The maximum permissible ***absolute discrepancy*** is defined as positive constant in the ABS parameter and forwarded to the function block as a number of digits (0 to 2048).
A parameter value of 0 disables the discrepancy check for absolute discrepancy.

One of six predefined values may be specified as maximum permissible ***relative discrepancy***. The required value is specified via the function block's REL parameter.

The following parameter specifications are used to select one of the predefined values:

REL parameter =KF+0 Discrepancy check for relative discrepancy disabled
REL parameter =KF+1 3.125%
REL parameter =KF+2 6.25 %
REL parameter =KF+3 12.5 %
REL parameter =KF+4 25 %
REL parameter =KF+5 50 %

The reference value for the relative discrepancy is the average of the two analog values.

It is not permitted to disable both discrepancy checks at the same time (ABS and REL). If an attempt is made to do so, FB232 immediately generates a discrepancy error.

Signal Standardization

A standardized signal is generated from the analog signals read.
Signal standardization is carried out in one of three specifiable modes.
The mode is set via the function block's NORM parameter.

NORM parameter =MA	Standardize to maximum value
NORM parameter =MI	Standardize to minimum value
NORM parameter =MW	Standardize to average value

Depending on this parameter, either the higher or lower of the two measured values or the average of the two is used for subsequent processing.

Scaling the Output Value

The standardized signal is scaled according to the OGR and UGR parameters and output to parameter XA.

The following applies to scaling:

OGR parameter	=	Output value for measured value 2048	(upper limit value)
UGR parameter	=	Output value for measured value 0	(lower limit value)

It is also possible to define a lower value for the upper limit than for the lower limit, which results in inverse behavior of the output value, i.e. an increase in the input signal means a decrease in the output signal.

Schematic for the Scaled Range -1000 to +1000

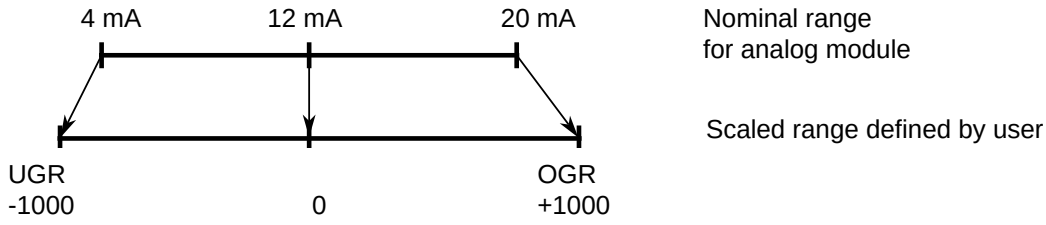


Figure 11-14. Scaling Schematic for a Scaling Range of -1000 to +1000

Response to a Discrepancy Error

In the event of a discrepancy error, a binary error signal (DISK) is generated at the function block's output.
This signal is 1 if there is no error and 0 if an error was detected.
The discrepancy error always affects this signal without a delay.

The discrepancy error also affects the standardized analog value at the block's output.

One of two different error responses can be preselected, depending on the VERH parameter at the block's input.

VERH parameter =SA	Immediate shutdown
VERH parameter =ZV	Time-restricted availability

In the case of ***immediate shutdown***, the output value is set to one of three possible values, depending on block parameter FEHL.

FEHL parameter =ER	Substitute value
FEHL parameter =MA	Maximum value
FEHL parameter =MI	Minimum value

The substitute value must be forwarded to FB 232 in the ERSA parameter as unscaled value in the range -512 to +3584.
Maximum and minimum value relate to the current, as yet unscaled measured input values.
The associated scaled measured value then appears at output XA, the corresponding unscaled measured value at output XA1.
If MA or MI is specified as error response, the output signal is corrected as per the analog input signals even after a discrepancy error has been detected.

In the case of ***time-restricted availability***, the last valid old value is forwarded to the function block's output for a specifiable period of time following detection of a discrepancy error.
Once this time period has elapsed, the block shows the same performance characteristics as for ***immediate shutdown***.

Time measurement for restricted availability is handled by a clock pulse which is initialized via the block's binary input TAKT.
The positive edges of the clock pulse are used for timing.
The time value is forwarded to the function block via the ZEIT parameter.
This value corresponds to the time value, in seconds, for restricted availability when TAKT is specified in seconds.
However, other timing units are also possible.

A discrepancy error is always latching.
The errored state is not cancelled until the signal is once again within the tolerance range ***and*** a positive edge is generated at the QUIT input.

FB 232 can identify the following error states, treating them in the same way as discrepancy errors:
- Wirebreak in an analog channel (current limit 3 mA)
- Over-range in an analog channel (measured value > 3583)
- Range violation at inputs XE1, XE2 (measured value < -512 or measured value > 3583)
- Neither relative nor absolute discrepancy check enabled (ABS and REL parameters are both 0)
- Values for OGR and/or UGR are not in range

Safety Note

Other than the resetting of the DISK signal and the output of substitute values, function block FB 232 gives no further response as regards a shutdown.
Any system-specific additional measures (such as PLC STOP, disabling actuators, etc.) must be included in the user program.

Diagnostic Flags

In addition to the signal at the DISK output, additional flags are set in the diagnostic byte in DR 4 of the programmed auxiliary DB.

Measured values outside the nominal range do not result in a discrepancy error as long as the module does not flag a range violation. However, a bit is set in the diagnostic byte when the standardized measured value is not in the nominal range.

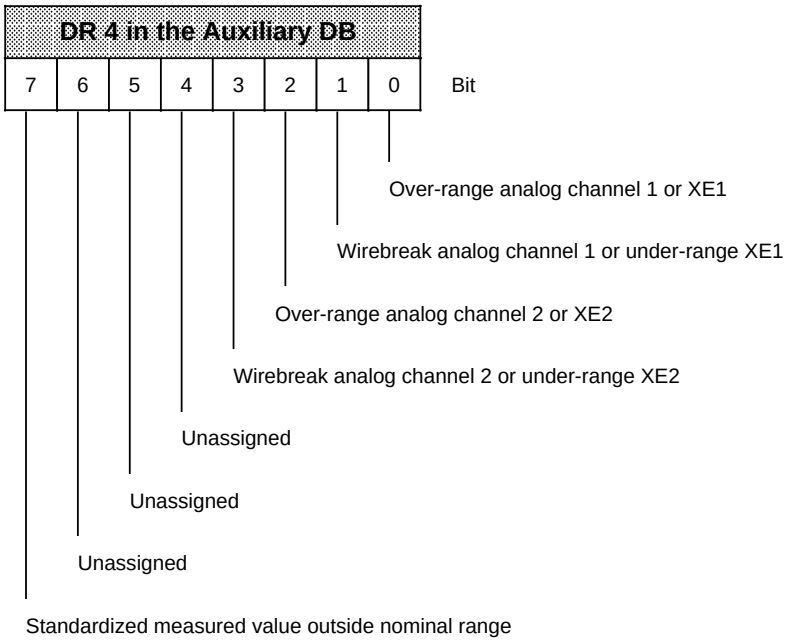


Figure 11-15. Diagnostic Byte in DR 4

11.7.2 Parameterization Example for FB 232

Task definition:
Two analog inputs are to be standardized to one analog signal through averaging.
In the range of small measured values, an absolute deviation of 50 units shall be permissible.
In the range of large measured values, a deviation of 12.5% from the average value shall be permissible.
In the event of a discrepancy of the analog inputs outside the permissible range, the block is to make the last valid value available at its output for 20 minutes, then output the substitute value for measured value 0.
Output signal XA is to be scaled for measured values 0 to 2048 in a range of 0 to 1000.

Program:

STL					Description
OB 1					E:BSP232ST.S5D
NETWORK 1			0000		
0000		:JU FB		232	Discrepancy evaluation for 2 analog inputs
0001	NAME	:RLG:AE2			
0002	BGK1	:	KY	0,0	Slot 0, channel 0 in subunit A
0003	BGK2	:	KY	1,0	Slot 1, channel 0 in subunit B
0004	OGR	:	KF	+1000	Scaled value for measured value 2048
0005	UGR	:	KF	+0	Scaled value for measured value 0
0006	XE1	:	DW	6	Not required here
0007	XE2	:	DW	7	Not required here
0008	DB	:	DB	232	Auxiliary data block
0009	TAKT	:	F	50.0	Flag in seconds
000A	ZEIT	:	KF	+1200	Time-restricted availability 1200 sec
000B	QUIT	:	I	2.0	Acknowledgement input
000C	NORM	:	KS	MW	In normal operation, output average value
000D	VERH	:	KS	ZV	In the event of an error, time-restricted availability
000E	FEHL	:	KS	ER	Output substitute value when ZV has elapsed
000F	ABS	:	KF	+50	Permissible absolute deviation
0010	REL	:	KF	+3	Permissible relative deviation 12.5%
0011	ERSA	:	KF	+0	Substitute value 0
0012	XA	:	DW	8	Standardized output signal
0013	DISK	:	F	50.1	Discrepancy flag
0014	XA1	:	DW	9	Non-standardized output signal
0015		:			
0016		:BE			

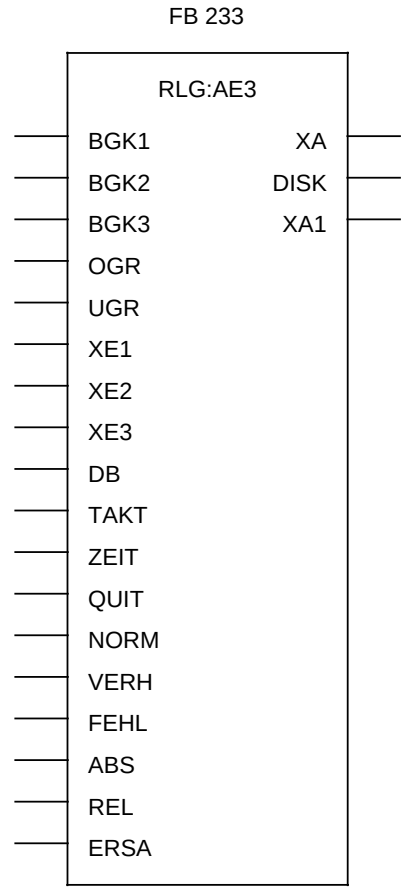
11.7.3 Discrepancy Analysis for Three Analog Inputs -FB 233-

FB 233 Block Parameters

STL

```
0000         :JU FB     233
0001  NAME    :RLG:AE3
0002  BGK1    :
0003  BGK2    :
0004  BGK3    :
0005  OGR     :
0006  UGR     :
0007  XE1     :
0008  XE2     :
0009  XE3     :
000A  DB      :
000B  TAKT    :
000C  ZEIT    :
000D  QUIT    :
000E  NORM    :
000F  VERH    :
0010  FEHL    :
0011  ABS     :
0012  REL     :
0013  ERSA    :
0014  XA      :
0015  DISK    :
0016  XA1     :
0017          :
0018          :BE
```

CSF



Note

The data block specified at input parameter DB is allocated to the three analog channels read (parameters BGK1, BGK2 and BGK3 or XE1, XE2 and XE3).
If you want to read three additional analog channels with FB 233, you must specify another data block.

Table 11-12. FB 233 Block Parameters

Parameter	Description	Type		Contents	Remarks
BGK1	Module slot/channel no. of 1st analog input	D	KY	KY = x,y x = 0 to 7 (slot) y = 0 to 3 (channel) Exception: KY=255,255 when using direct block input XE1	
BGK2	Module slot/channel no. of 2nd analog input	D	KY	KY = x,y x = 0 to 7 (slot) y = 0 to 3 (channel) Exception: KY=255,255 when using direct block input XE2	
BGK3	Module slot/channel no. of 3rd analog input	D	KY	KY = x,y x = 0 to 7 (slot) y = 0 to 3 (channel) Exception: KY=255,255 when using direct block input XE3	
OGR	Upper limit of scaled range	D	KF	-8192 to +8192	
UGR	Lower limit of scaled range	D	KF	-8192 to +8192	
XE1	Direct block input for an unscaled analog value in place of 1st analog channel BGK1	I	W	FW 0 to FW 198 DW6 to DW255 Value range: -512 to +3583	*1), *2)
XE2	Direct block input for an unscaled analog value in place of 2nd analog channel BGK2	I	W	Same as parameter XE1	*1), *2)
XE3	Direct block input for an unscaled analog value in place of 3rd analog channel BGK3	I	W	Same as parameter XE1	*1), *2)

Table 11-12. FB 233 Block Parameters (continued)

Parameter	Description	Type		Contents	Remarks
DB	Auxiliary data block for storing old values, error states and user data	B		DB 2 to DB 251 Minimum length 10 DW	
TAKT	Clock pulse for time-restricted availability	I	BI	I/Q/F bit, e.g. in seconds	
ZEIT	Time, in TAKT units, for time-restricted availability	D	KF	0 to +32767	
QUIT	Acknowledgement input for acknowledging errors	I	BI	I/Q/F bit	
NORM	Selection for generating the output signal in normal operation	D	KS	MW = Average value MA = Maximum value MI = Minimum value	
VERH	Selection for response to a discrepancy error	D	KS	SA = Immediate shutdown ZV = Time-restricted availability UV = Unrestricted availability	
FEHL	Selection for generating the output signal when a discrepancy error occurs	D	KS	ER = Substitute value MA = Maximum value MI = Minimum value	
ABS	Permissible absolute error tolerance based on the as yet unscaled input value	D	KF	0 to +20480 = disabled	

Table 11-12. FB 233 Block Parameters (continued)

Parameter	Description	Type		Contents	Remarks
REL	Permissible relative error tolerance based on the unscaled input value	D	KF	0 = Disabled 1 = 3.125% 2 = 6.25% 3 = 12.5% 4 = 25% 5 = 50%	
ERSA	Substitute value as unscaled input value	D	KF	-512 to +3584	
XA	Block output for standardized and scaled measured value	Q	W	FW0 to FW198 DW10 to DW254 QW	*1)
DISK	Output bit for discrepancy error (no error = 1)	Q	BI	I/Q/F bit	
XA1	Output of standardized unscaled measured value	Q	W	FW0 ...FW198 DW10...DW255 QW	*1)

*1) If data words are used for these parameters, they must lie in the data block specified in the DB parameter. Data words 0 to 9 are reserved internally by the function block, and may not be used for this purpose. If the parameter is not needed, simply specify a free data word in the data block.

*2) A discrepancy is flagged when the permissible value range is exceeded.

Execution Time and Operands Reserved by FB 233

FB 233's execution time is approximately 7.5 ms. FB 233 reserves the following operands:

Table 11-13. Reserved Operands

Type	Reserved Area
Temporary flags	FY 240 to FY 255
Data words	DW 0 to DW 9 in the specified DB

FB 233 Functionality

The block can perform the functions listed below, which will be discussed in more detail in the following.

- Reading of three analog signals
- Discrepancy analysis
- Signal standardization
- Scaling of the output signal
- Response to discrepancy errors

Reading of Analog Signals

As standard function, the block reads three analog channels.
Specification of the module slot and the channel number, which are both part of the BGK1, BGK2 and BGK3 parameters, suffices to define the address of the analog signals to be read.

Discrepancy Analysis

A discrepancy analysis is carried out for the three analog values read.
This includes checking for a maximum permissible ***absolute*** discrepancy ***and*** a maximum permissible ***relative*** discrepancy.
A discrepancy error is detected when the discrepancy in the three analog values has exceeded both the absolute and the relative limit value.

The maximum permissible ***absolute discrepancy*** is forwarded to the function block as a number of digits (0 to 2048) and specified as positive constant in the ABS parameter.
The check for absolute discrepancy is disabled in the case of parameter 0.

One of six predefined values can be specified for the maximum permissible ***relative discrepancy***.
The specification is made via the function block's REL parameter.

The following applies for the predefined values:

REL parameter =KF+0	Discrepancy check for relative discrepancy disabled
REL parameter =KF+1	3.125%
REL parameter =KF+2	6.25 %
REL parameter =KF+3	12.5 %
REL parameter =KF+4	25 %
REL parameter =KF+5	50 %

The reference value for the relative discrepancy is the average of the three analog values.

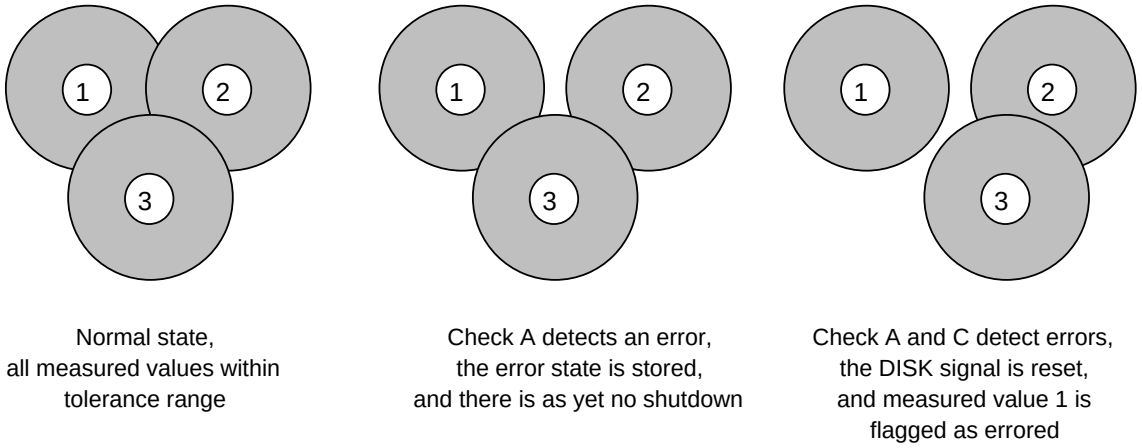
Disabling both discrepancy checks (ABS and REL) is not allowed.
If an attempt is made to do so, FB233 immediately generates a discrepancy error.

Principle of the Discrepancy Checks

In a discrepancy check, all three measured values are always compared to one another in pairs.

Check A: = Compares analog value 1 with analog value 2
Check B: = Compares analog value 2 with analog value 3
Check C: = Compares analog value 1 with analog value 3

A measured value is always flagged as errored and phased out of the processing cycle when its tolerance limit to the other two measured values is exceeded.



Normal state, all measured values within tolerance range

Check A detects an error, the error state is stored, and there is as yet no shutdown

Check A and C detect errors, the DISK signal is reset, and measured value 1 is flagged as errored

= Tolerance range of the measured values

1 2 3 = Measured values 1 to 3

Figure 11-16. Discrepancy Test for the Three Measured Values

Signal Standardization

A uniform signal is generated from the analog signals read.
Signal standardization can be performed in one of three modes.
The required mode is set via the function block's NORM parameter.

NORM parameter =MA	Standardize to maximum value
NORM parameter =MI	Standardize to minimum value
NORM parameter =MW	Standardize to average value

Depending on this parameter, either the larger or the smaller of the three measured values or the average value of all three measured values is used for subsequent processing.

Scaling the Output Value

The standardized signal is scaled as per the OGR and UGR parameters and output to parameter XA.

The following always applies:

OGR parameter	=	Output value for measured value 2048	(upper limit)
UGR parameter	=	Output value for measured value 0	(lower limit)

It is also possible to specify a lower value for the upper limit than for the lower limit, which results in inverse behavior of the output value, i.e. an increase in the input signal means a decrease in the output signal.

Schematic for the Scaled Range of -1000 to +1000

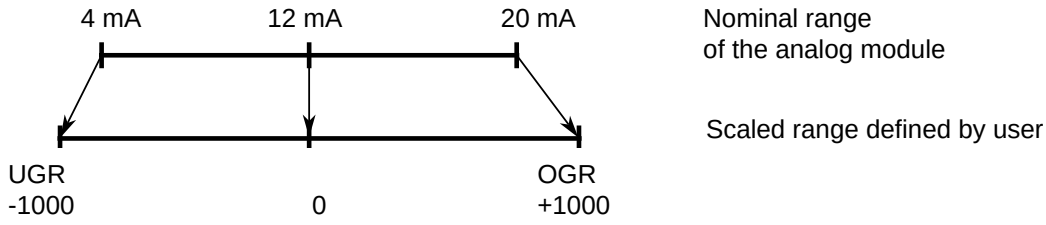


Figure 11-17. Scaling Schematic for a Scaling Range of -1000 to +1000

Response to a Discrepancy Error

When a discrepancy error is detected, a binary error signal (DISK) is generated at the function block's output.
This signal is 1 when there is no error, and 0 when a discrepancy error was detected.
The discrepancy error always affects this signal without a delay.

The discrepancy error also affects the standardized analog value at the block's output.

Depending on the VERH parameter at the block's input, the user may choose between two different error responses.

VERH parameter =SA	Immediate shutdown
VERH parameter =ZV	Time-restricted availability
VERH parameter =UV	Unrestricted availability

In the event of ***immediate shutdown***, the output value is immediately set to one of three possible values in dependence on the block's FEHL parameter.

FEHL parameter =ER	Substitute value
FEHL parameter =MA	Maximum value
FEHL parameter =MI	Minimum value

The substitute value must be forwarded to FB 233 via the ERSA parameter as unscaled value in the range from -512 to +3584.
Maximum value and minimum value refer to the current, as yet unscaled measured input values.
The associated scaled measured value then appears at output parameter XA, the corresponding unscaled value at output parameter XA1.
When MA or MI is specified as error response, the output signal is corrected as per the analog input signals, even after detection of a discrepancy error.

In the case of ***time-restricted availability***, the last valid value is forwarded to the function block's output for a specifiable period of time when a discrepancy error occurs.
When this time period has elapsed, the block shows the same performance characteristics as for ***immediate shutdown***.

The period for time-restricted availability is programmed via the ZEIT and TAKT parameters. Specify the number of units of time with the ZEIT parameter and the duration of one unit with the TAKT parameter.
Example: The value 12000 was programmed in the ZEIT parameter, and TAKT was programmed for seconds. Time-restricted availability is thus for a period of 12000 seconds.

In the case of ***unrestricted availability with regard to time***, the standardized value (average, maximum or minimum value) of the two analog signals that are still correct are forwarded to the block's output for an unrestricted period of time when a discrepancy error occurs.
If a second error occurs, the block shows the same performance characteristics as for ***immediate shutdown***.

A discrepancy error is always latching. The error status is not revoked until the signals once again lie within the tolerance range ***and*** a positive edge is generated at the QUIT input.

FB 233 also checks for the following error states and treats them as though they were discrepancy errors:
- Wirebreak in an analog channel (current limit 3 mA)
- Over-range, analog channel (measured value > 3583)
- Range violation at inputs XE1, XE2, XE3 (measured value < -512 or measured value > 3583)
- Neither relative nor absolute discrepancy check enabled (block parameters ABS and REL are both 0).
- Values for OGR and/or UGR are not within the permissible range.

Safety Note

In addition to the resetting of the DISK signal and the output of substitute values, function block FB 233 does not initiate any shutdown functions.
Any additional system-specific measures (such as PLC STOP, disabling actuators, etc.) must be included in the user program.

Diagnostic Flags

In addition to the DISK signal, certain errors also set flags in diagnostic word DW15 of the programmed auxiliary DB.
A bit value of '1' is indicative of an error.
Measured values outside the nominal range do not result in a discrepancy error as long as the module has not yet flagged a range violation. However, a bit is set in the diagnostic byte when the standardized measured value is outside the nominal range.

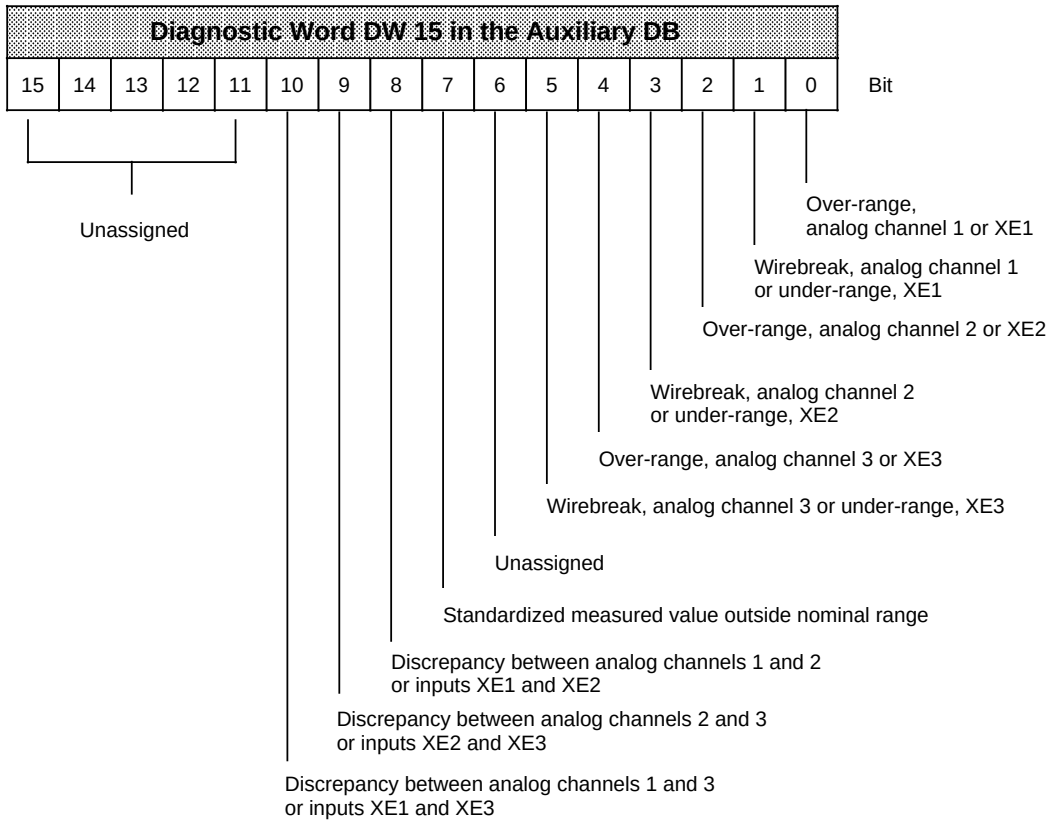


Figure 11-18. Diagnostic Byte in DW 15

Parameterization Example for FB 233

Three analog inputs are to be standardized through averaging.
In the range of small measured values, an absolute discrepancy of 40 units is to be permitted.
In the range of large measured values, a discrepancy of 25% from the average is to be permitted.
Should an analog signal fail, the block is to continue executing for a maximum of four hours with the two remaining input signals.
If the four hours elapse without the error being rectified and acknowledged, or when a second error occurs, the substitute value for measured value 2048 is to be forwarded to the output.
Output signal XA is to be scaled for measured values 0 to 2048 in the range from 1000 to 5000.

Program:

STL	Description
OB 1	E:BSP233ST.S5D
NETWORK 1 0000	
0000 :JU FB 233	Discrepancy evaluation for 3 analog inputs
0001 NAME :RLG:AI3	
0002 BGK1 : KY 0,0	Slot 0, channel 0 in subunit A
0003 BGK2 : KY 1,1	Slot 1, channel 0 in subunit B
0004 BGK2 : KY 2,2	Slot 2, channel 0 in subunit A
0005 OGR : KF +5000	Scaled value for measured value 2048
0006 UGR : KF +1000	Scaled value for measured value 0
0007 XE1 : DW 16	Not required here
0008 XE2 : DW 17	Not required here
0009 XE3 : DW 18	Not required here
000A DB : DB 233	Auxiliary data block
000B TAKT : F 50.0	Flag in seconds
000C ZEIT : KF +14400	Time-restricted availability 14400 sec
000D QUIT : I 2.0	Acknowedgment input
000E NORM : KS MW	In normal operation, output average value
000F VERH : KS ZV	In event of error, time-restricted availability
0010 FEHL : KS ER	When ZV has expired, output substitute value
0011 ABS : KF +40	Permissible absolute deviation
0012 REL : KF +4	Permissible relative deviation 25%
0013 ERSA : KF +2048	Substitute value 2048
0014 XA : DW 18	Scaled output signal
0015 DISK : F 50.2	Discrepancy flag
0016 XA1 : DW 19	Unscaled output signal
0017 :	
0018 :BE	

11.7.4 Combining FB 232 and FB 233

Function blocks FB 232 anad FB 233 can also be combined in the user program. In this way, it is possible to standardize more than three analog signals into a single analog value.

When combining these blocks, be sure that the required auxiliary data blocks have different block numbers.

Input parameters XE1 to XE3 and output parameter XA1 are used to implement this function.

Inputs XE1 to XE3 always process unscaled analog values, and output XA1 generates an unscaled analog value.

When the blocks are combined, the XA1 outputs of the preceding blocks must be brought into contact with inputs XE1 to XE3 of the following block.

The standardized scaled signal can then be picked up at the XA output of the last block.

To monitor for discrepancy errors, the DISK outputs of all combined blocks must, as a rule, be evaluated.

11.8 Circuit Versions for Function Blocks FB 232 and FB 233

Failsafe analog value processing is possible when the analog signals are read in with the 464-8MG11 analog module and evaluated with the aid of integral function blocks FB 232 and FB 233.

As per quality levels to DIN V 19250, two or more analog modules must be used and those used must be distributed between two subunits. All associated analog signals must be placed on different 464-8MG11 AI modules.

Quality Levels to DIN V 19250 and Characteristics of the Analog Signals

Depending on the circuit version, one differentiates between several I/O types in S5-95F systems (see sections 11.8.1 to 11.8.8). The attainable quality level depends on the time it takes for the analog signals to change. The analog signals can be subdivided into:

- ***static analog signals***, which change only rarely, if at all, during normal operation, and
- ***dynamic analog signals***, which change frequently during operation or pass through the operational, safety-related value range within 24 hours.
  An analog signal is also dynamic when it is intermittent at the user level.

To make an analog signal intermittent at the user level, you must, for example, interrupt it at regular intervals. After a waiting time of 500 ms, the analog input signals that were read in must produce a value of "0", which otherwise will not occur during normal operation (the nominal range is 4 to 20 mA).
When the interruption is revoked, the current analog value can once again be read in/post-processed in the user program after another 500 ms have elapsed.

To find out which circuit versions are suitable or permissible for your application, you must consult with the authorized technical expert.

Table 11-14. Quality Levels for Failsafe Analog Input Modules

I/O Type	Section	Quality Level to DIN V 19250 for ...	
		Dynamic Signals	Static Signals
Type R 4.2	Section 11.8.1	AK4	AK3
Type R 4.4	Section 11.8.2	AK4	AK4
Type R 5.1	Section 11.8.3	AK5	AK3
Type R 6.1	Section 11.8.4	AK6	AK3
Type R 6.2	Section 11.8.5	AK6	AK5
Type R 6.3	Section 11.8.6	AK6	AK5
Type R 6.4	Section 11.8.7	AK6	AK4
Type R 6.5	Section 11.8.8	AK6	AK6

Safety Note

Detailed expert knowledge regarding safety requirements and the approach to errors, in addition to thorough knowledge standards DIN V 19250, DIN V 19251, DIN V VDE 0801 and/or IEC 1508 are needed in order to process static analog signals.

Circuit Versions for Use in the Domain of the German Federal Bureau of Railways (EBA)

Failsafe analog input modules that are to be used within the purview of the EBA must satisfy a special error model. The I/O types approved for this area of application are listed in a separate certification report. Please direct your questions to the address given in Appendix E.

Requirements Placed on Analog Sensors

The circuit versions shown are based on ***conventional, field-tested sensors with minimum complexity***. If you use other sensors, you must take appropriate measures to prevent systematic sensor errors (e.g. diversity-based sensors).
When selecting sensors, note that **non-failsafe** analog sensors require **diversity** at the very least.

Requirements Placed on the Supply of Analog Sensors

The 464-8MG11 analog input module does not provide voltage to supply analog sensors. For this reason, supply your analog sensors from the 24 V DC source used to supply the base unit and the AI module. The sensors must function error-free within the permissible voltage limits.

Function Test for AI 464-8MG11 at the User Level

The 464-8MG11 analog input module has no integral test equipment (as does, for instance, the DI 431-8FA11). For this reason, you must subject the analog input modules with failsafe requirements to a function test at regular intervals. These intervals depend on the I/O type.

The table below lists the I/O types and the intervals for the function test.

Table 11-15. Intervals for the AI 464-8MG11 Function Test at the User Level

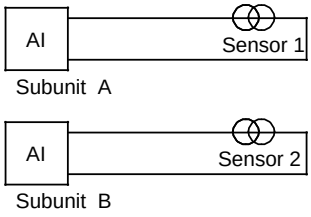
I/O Type	Section	Function Test Required At Least Every ...
Type R 4.2	Section 11.8.1	30 months
Type R 4.4	Section 11.8.2	5 months
Type R 5.1	Section 11.8.3	3 months
Type R 6.1	Section 11.8.4	2 months
Type R 6.2	Section 11.8.5	12 months
Type R 6.3	Section 11.8.6	1.5 months
Type R 6.4	Section 11.8.7	6 months
Type R 6.5	Section 11.8.8	12 months

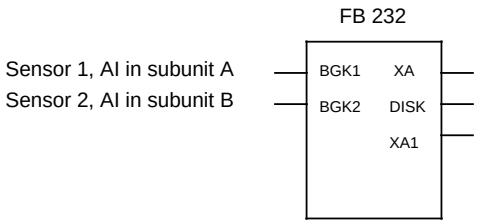
When you perform a function test, you must make sure that the S5-95F initiates the safety response expected (for the I/O type being tested).

Example: You measure the level of a tank and use failsafe monitoring equipment to monitor the maximum permissible level. When the maximum level is exceeded, the S5-95F's safety response is supposed to close the intake valve.
During the function test, you must make sure that the S5-95F initiates the safety response when "maximum level exceeded" is reported. It is not necessary to verify the level or the S5-95F's method of operation within the acceptance range when there are no errors, as (in this example) no safety response will be derived therefrom.

11.8.1 Circuit Type R4.2 for AK4

Table 11-16. Circuit Diagram for Type R4.2 for AK4

Circuit Diagram	Characteristics
I/O type R4.2 	Evaluation of sensors: 2 out of 2 Evaluation of analog inputs: 2 out of 2 Quality level: AK4 to DIN V 19250 for dynamic sensor values; AK3 to DIN V 19250 for static analog sensor values Availability after error: None Function block used: FB232 Number of data blocks required: 1

Interconnection of Blocks	Block Parameters
	FB 232: NORM = MW,MI,MA VERH = SA FEHL = ER,MA,MI

Circuit Performance on Error/Required User Response

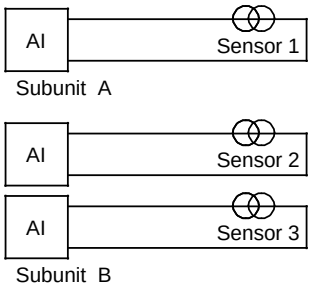
Performance upon detection of a sensor failure:
The DISK signal is reset and the value specified for this error situation is forwarded to the block output.
Performance upon detection of an analog input failure:
Same as for a sensor failure.
User response:
The evaluation of the DISK signal or the output value programmed for this error situation must lead to shutdown of the relevant part of the installation or to a safe state.

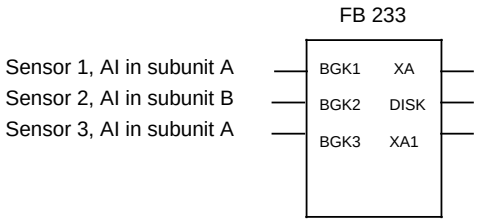
Safety Note

The AI 464-8MG11 module has no integral test facility. For this reason, a function test at the user level is necessary for the selected I/O type. You will find a list of the intervals at which the test must be conducted for each I/O type in Table 11-15.

11.8.2 Circuit Type R4.4 for AK4

Table 11-17. Circuit Diagram for Type R4.4 for AK4

Circuit Diagram	Characteristics
I/O type R4.4 	Evaluation of sensors: 3 out of 3/2 out of 3 Evaluation of analog inputs: 3 out of 3/2 out of 3 Quality level: AK4 to DIN V 19250 for dynamic and static analog sensor values Availability after error: Unrestricted Function block used: FB233 Number of data blocks required: 1

Interconnection of Blocks	Block Parameters
	FB 233: NORM = MW,MI,MA VERH = SA,ZV,UV FEHL = ER,MA,MI

Circuit Performance on Error/Required User Response:

Performance upon detection of a sensor failure:
The DISK signal is reset. If the mode is VERH = ZV or UV, the block continues to execute for a limited time or for an unlimited time with the two analog inputs that are still functioning. If a second error is detected or if the mode is VERH = SA, the value specified for the error situation is forwarded to the block's output.

Performance upon detection of an analog input failure:
Same as for a sensor failure.

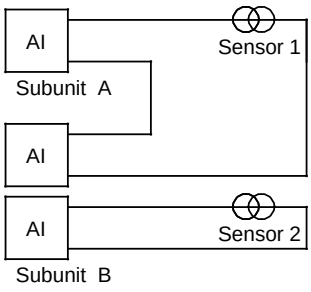
User response:
In mode VERH = ZU/UV, the DISK signal must trigger an alarm. The output value specified for the error situation must result in a shutdown of the relevant part of the installation or to a safe state.
If a 3-out-of-3 evaluation is required, the mode must be set to VERH = SA. In this case, the evaluation of the DISK signal or the output value specified for the error situation must lead to a shutdown of the relevant part of the installation or to a safe state.

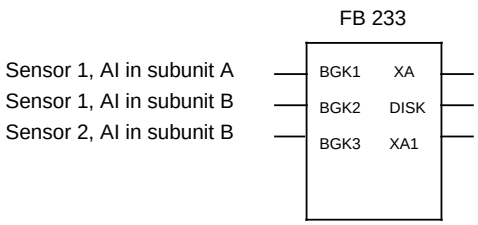
Safety Note

The AI 464-8MG11 module has no integral test facility. For this reason, a function test at the user level is required for the selected I/O type. The interval at which the test must be carried out can be found in Table 11-15.

11.8.3 Circuit Type R51 for AK5

Table 11-18. Circuit Diagram for Type R5.1 for AK5

Circuit Diagram	Characteristics
I/O type R5.1  	Evaluation of the sensors: 2 out of 2 Evaluation of the analog inputs: 3 out of 3 Quality level: AK5 to DIN V 19250 for dynamic analog sensor values; AK3 to DIN V 19250 for static analog sensor values Availability after error: None Function block used: FB233 Number of data blocks required: 1

Interconnection of Blocks	Block Parameters
 	FB 233: NORM = MW,MI,MA VERH = SA FEHL = ER,MA,MI

Circuit Performance on Error/Required User Response

Performance upon detection of a sensor failure:
The DISK signal is reset and the value specified for the error situation is forwarded to the block's output.

Performance upon detection of an analog input failure:
Same as for a sensor failure.

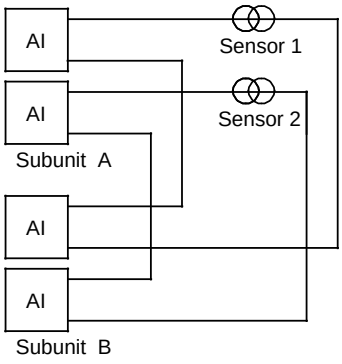
User Response:
The evaluation of the DISK signal or the output value for this error situation must lead to a shutdown of the relevant part of the installation or to a safe state.

11.8.4 Circuit Type R6.1 for AK6

Table 11-19. Circuit Diagram for Type R6.1 for AK6

Circuit Diagram

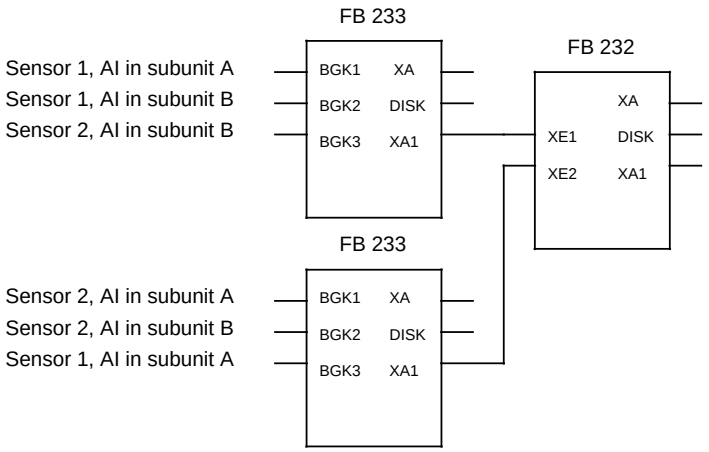
I/O type R6.1



Characteristics

Evaluation of sensors:	2 out of 2
Evaluation of the analog inputs:	3 out of 4
Quality level:	AK6 to DIN V 19250 for dynamic analog sensor values; AK3 to DIN V 19250 for static analog sensor values
Availability after sensor error:	None
Availability after AI error:	Unrestricted
Function blocks used:	FB233/FB232
Number of data blocks required:	3

Interconnection of Blocks



Block Parameters

FB 233:
NORM = MW,MI,MA
VERH = ZV/UV
FEHL = ER
ERSA = 3584
FB 232:
NORM = MW,MI,MA
VERH = SA
FEHL = ER
ERSA = xxxx

Table 11-19. Circuit Diagram for Type R6.1 for AK6 (continued)

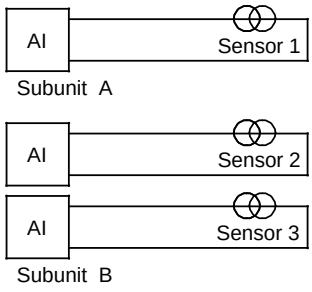
Circuit Performance on Error/Required User Response
Performance upon detection of a sensor failure: The DISK signal for block FB233 is reset. Output of the substitute value +3584 also resets the DISK signal of the following block FB232 and outputs the substitute value specified there. **Performance upon detection of an analog input failure:** The DISK signal for block FB233 is reset. Depending on the mode setting (VERH = ZV/UV), function block FB233 then executes for a limited time or for an unlimited time with the two remaining analog inputs. The DISK signal at FB 232's output is not affected. If a second error is detected or when the time-restricted availability period has elapsed, the circuit's performance is the same as for a sensor failure. **User Response:** The DISK signals of the preceding FB233s must trigger an alarm. The evaluation of FB232's DISK signal or the output value specified for this error situation must lead to a shutdown of the affected part of the installation or to a safe state.

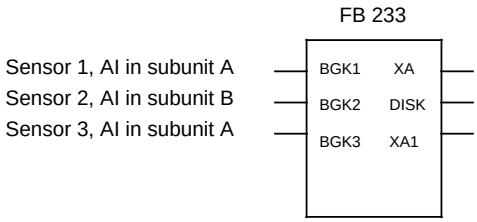
Safety Note

- The AI 464-8MG11 module has no integral test facility. For this reason, a function test is required at the user level for the selected I/O type. The intervals at which this test is required can be found in Table 11-15.
- In the mode with time-restricted availability, the length of the time period is determined by the multiple error occurrence time (in burner management systems, for example, 24 hours).
- When the blocks are used in "averaging" mode (NORM = MW), it must be taken into account that, because of special wiring, analog inputs AI1.A and AI2.B are evaluated twice in the calculation of the average value.

11.8.5 Circuit Type R6.2 for AK6

Table 11-20. Circuit Diagram for Type R6.2 for AK6

Circuit Diagram	Characteristics
I/O type R6.2 	Evaluation of sensors: 3 out of 3 Analog inputs: 3 out of 3 Quality level: AK6 to DIN V 19250 for dynamic analog sensor values; AK5 to DIN V 19250 for static analog sensor values Availability after error: None Data block used: FB233 Number of data blocks required: 1

Interconnection of Blocks	Block Parameters
	FB 233: NORM = MW,MI,MA VERH = SA FEHL = ER,MA,MI

Circuit Performance on Error/Required User Response

Performance upon detection of a sensor failure:
The DISK signal is reset and the value programmed for the error situation is forwarded to the block's output.

Performance upon detection of an analog input failure:
The same as for a sensor failure.

User Response:
The evaluation of the DISK signal or the output value programmed for the error situation must lead to the shutdown of the relevant part of the installation or to a safe state.

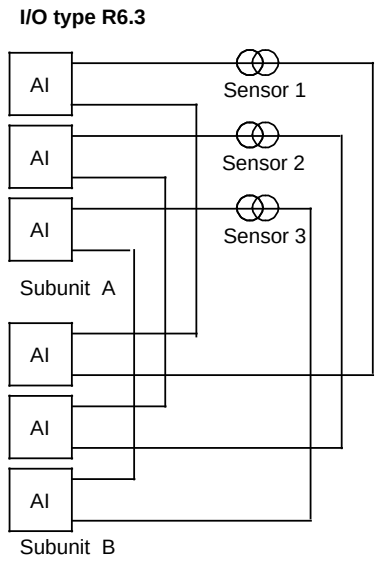
Safety Note

- The AI 464-8MG11 module has no integral test facility. For this reason, a function test is required at the user level for the selected I/O type. The intervals at which the test must be carried out are given in Table 11-15.
- In the mode with time-restricted availability, the permissible period of time is determined by the multiple error occurrence time (in burner management systems, for instance, 24 hours).

11.8.6 Circuit Type R6.3 for AK6

Table 11-21. Circuit Diagram for Type R6.3 for AK6

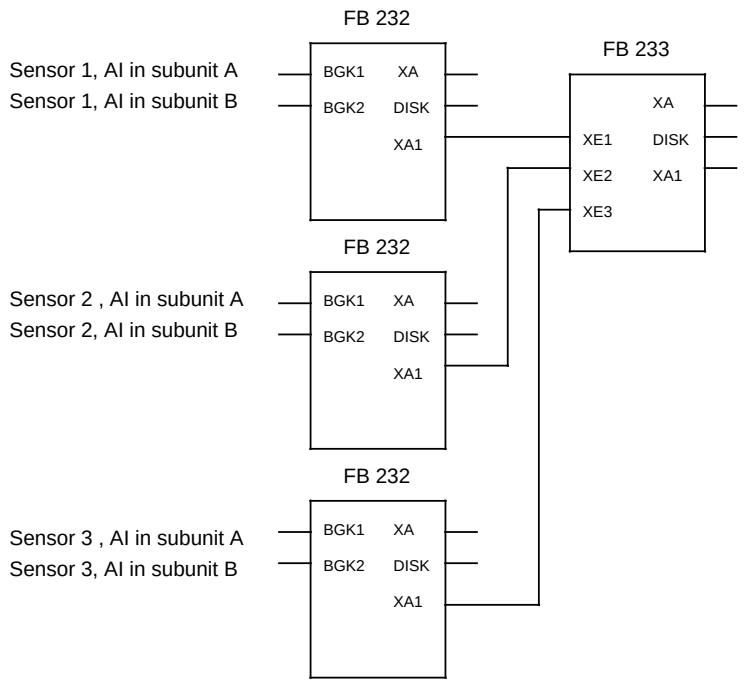
Circuit Diagram



Characteristics

Evaluation of sensors:	2 out of 3
Evaluation of analog inputs:	2 out of 3 (2 out of 2)
Quality level:	AK6 to DIN V 19250 for dynamic analog sensor values; AK5 to DIN V 19250 for static analog sensor values
Availability after error:	Time-restricted
Function blocks used:	FB232,FB233
Number of data blocks required:	4

Interconnection of Blocks



Block Parameters

FB 232:

NORM	= MW,MI,MA
VERH	= SA
FEHL	= ER
ERSA	= 3584

FB 233:

NORM	= MW,MI,MA
VERH	= ZV
FEHL	= ER
ERSA	= xxxx

Table 11-21. Circuit Diagram for Type R6.3 for AK6 (continued)

Circuit Performance on Error/Required User Response
Performance upon detection of a **sensor** failure: The associated FB233 input is marked errored and phased out. The DISK signal for FB233 is reset. Function block FB233 then executes for a limited time with the remaining two analog inputs. If a second error is detected or when the time-restricted availability period has elapsed, the value programmed for the error situation is forwarded to FB233's output. **Performance upon detection of an analog input failure:** The DISK signal for FB232 is reset. Output of the substitute value +3584, the relevant input in the following function block FB233 is marked errored and phased out of the processing cycle. Otherwise, performance is the same as for a sensor error. **User response:** FB233's DISK signal must trigger an alarm. The FB233 output value programmed for the error situation must lead to a shutdown of the affected part of the installation or to a safe state.

Safety Note

- The AI 464-8MG11 module has no integral test facility. For this reason, a function test is required for the selected I/O type. The intervals at which the test must be performed are given in Table 11-15.
- In the mode with time-restricted availability, the permissible time period is determined by the multiple error occurrence time (in burner management systems, for example, 24 hours).

11.8.7 Circuit Type R6.4 for AK6

Table 11-22. Circuit Diagram for Type R6.4 for AK6

Circuit Diagram	Characteristics
I/O type R6.4 AI — Sensor 2 AI — Sensor 2 Subunit A AI — Sensor 3 AI — Sensor 4 Subunit B	Evaluation of sensors: 3 out of 4 Evaluation of analog inputs: 3 out of 4 Quality level: AK6 to DIN V 19250 for dynamic analog sensor values; AK4 to DIN V 19250 for static analog sensor values Availability after error: Unrestricted Function blocks used: FB233,FB232 Number of data blocks required: 3

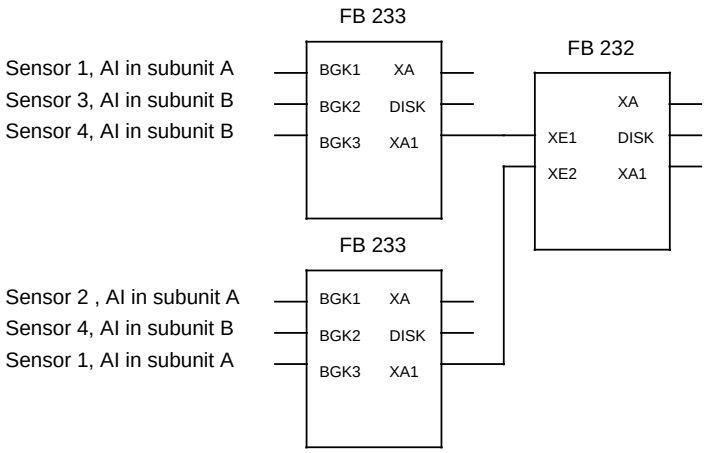
Interconnection of Blocks	Block Parameters
	FB 233: NORM = MW,MI,MA VERH = ZV/U FEHL = ER ERSA = 3584 FB 232: NORM = MW,MI,MA VERH = SA FEHL = ER ERSA = xxxx

Table 11-22. Circuit Diagram for Type 6.4 for AK6 (continued)

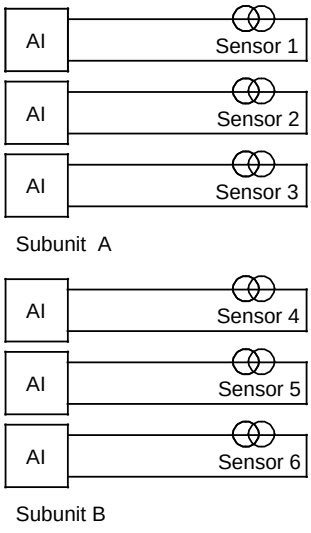
Circuit Performance on Error/Required User Response
Performance upon detection of a sensor failure: FB233's DISK signal is reset. Depending on the mode (VERH = ZV/UV), function block FB233 continues to execute for a limited time or for an unlimited time with the two remaining analog inputs. The DISK signal at FB 232's output is not affected. If a second error is detected or when the time-restricted availability period has elapsed, output of the substitute value +3584 resets the DISK signal of the following block FB232 and outputs the programmed substitute value there. **Performance upon detection of an analog input failure:** Same as for a sensor failure. **User response:** The DISK signals of the preceding FB233s must trigger an alarm. The evaluation of FB232's DISK signal or the output value programmed for the error situation must lead to a shutdown of the affected part of the installation or to a safe state.

Safety Note

- The AI 464-8MG11 module has no integral test facility. For this reason, a function test is required at the user level for the selected I/O type. The intervals at which the test must be performed are shown in Table 11-15.
- In the mode with time-restricted availability, the permissible time period is determined by the multiple error occurrence time (in burner management systems, for instance, 24 hours).
- In the mode with unrestricted availability, a signal change is required within the MFEZ. As a rule, this is ensured through intermittent operation.
- If the blocks are used in "averaging" mode (NORM = MW), it must be taken into account that, because of special wiring, analog inputs AI1.A and AI4.B are evaluated twice when calculating the average value.

11.8.8 Circuit Type R6.5 for AK6

Table 11-23. Circuit Diagram for Type R6.5 for AK6

Circuit Diagram	Characteristics
I/O type R6.5  	Evaluation of sensors: 2 out of (2 out of 3) Evaluation of analog inputs: 2 out of (2 out of 3) Quality level: AK6 to DIN V 19250 for dynamic and static analog sensor values Availability after error: Unrestricted Function blocks used: FB233, FB232 Number of data blocks required: 3

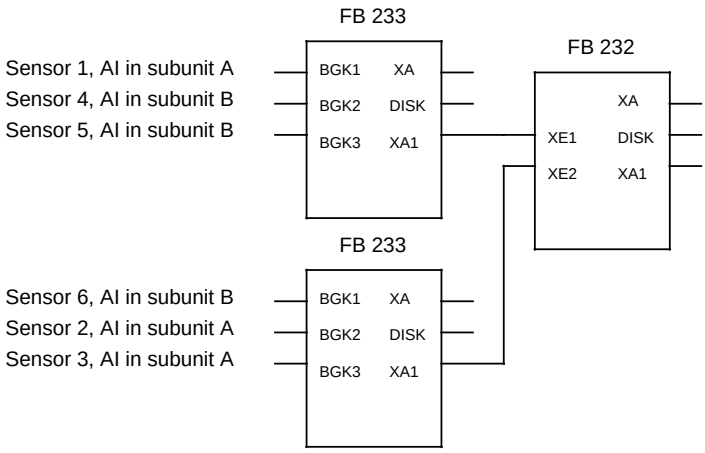
Interconnection of Blocks	Block Parameters
 	FB 233: NORM = MW,MI,MA VERH = UV FEHL = ER,MA,MI FB 232: NORM = MW,MI,MA VERH = SA FEHL = ER ERSA = xxxx

Table 11-23. Circuit Diagram for Type R6.5 for AK6 (continued)

Circuit Performance on Error/Required User Response
Performance upon detection of a sensor failure: Function block FB233's DISK signal is reset. FB233 then continues executing for an unrestricted amount of time with the remaining analog inputs. The DISK signal at FB232's output is not affected. If a second error is detected in the same FB233, output of the substitute value +3584 resets the DISK signal of the following FB232 and outputs the substitute value programmed there. **Performance upon detection of an analog input failure:** Same as for a sensor error. **User Response:** The DISK signals of the preceding FB233s must trigger an alarm. The evaluation of FB232's DISK signal or the output value programmed for the error situation must lead to a shutdown of the affected part of the installation or to a safe state.

Safety Note

- The AI 464-8MG11 module has no integral test facility. For this reason, a function test at the user level is required for the selected I/O type. The interval at which the test must be carried out is given in Table 11-15.
- In the mode with time-restricted availability, the permissible time period is determined by the multiple error occurrence time (in burner management systems, for instance, 24 hours).

12 Interrupt Processing

12.1 Using Onboard Interrupt Inputs 12- 1

12.2 Asynchronous Interrupt Processing in OB2 . . . . . 12- 2
12.2.1 Programming OB2 . . . . . 12- 4
12.2.2 Programming Interrupt Responses in OB2 . . . . . 12- 6
12.2.3 Connecting Interrupt Inputs 12- 7

12.3 Synchronous Interrupt Processing in OB3 . . . . . 12- 9
12.3.1 Programming OB3 . . . . . 12- 12
12.3.2 Programming Interrupt Responses in OB3 . . . . . 12- 13

12.4 Interrupt Response Times for the S5-95F 12- 14

Figures

12-1 Connecting a Failsafe Sensor to an Interrupt Input 12- 7
12-2 Connecting a Failsafe Sensor and a Line Monitor to an Interrupt Input 12- 7
12-3 Connecting Two Independent Sensors to an Interrupt Input 12- 8
12-4 Connecting Two Independent Sensors and a Line Monitor to an Interrupt Input 12- 8

Tables

12-1 Priorities of the Organization Blocks 12- 1
12-2 Differences Between OB2 and OB3 Interrupts 12- 1
12-3 Pin Assignments for the 9-Pin Sub D Connector 12- 2
12-4 Operations Permissible in OB2 . . . . . 12- 4
12-5 Communication and Diagnostic Bytes for Responses to OB2 Interrupts . . . 12- 5
12-6 Example for Programming OB2 . . . . . 12- 6
12-7 Example for Scanning the Diagnostic Bytes to Determine the Cause of an Interrupt 12- 13

12 Interrupt Processing

This section covers the following:

- Interrupt inputs on the S5-95F
- How interrupt inputs are connected
- How the characteristics for interrupt inputs are defined in DB1
- What happens "internally" during interrupt processing
- How to compute the response times to process interrupts

12.1 Using Onboard Interrupt Inputs

Program scanning should be interrupt-driven whenever a particularly fast response to changes in process signals is required.
A change in the process signal at an interrupt input interrupts cyclic program scanning (OB1), time-controlled program scanning (OB13) or the operating system and starts an interrupt service routine (OB2 or OB3). When the interrupt service routine has executed, the S5-95F returns to the point of interruption and program scanning is resumed.

The onboard inputs to be used as interrupt inputs are specified in DB1 as part of the process of initializing the onboard I/Os.

In the DB1, you can initialize up to

- 4 interrupt inputs for triggering OB2 as interrupt service routine and
- 20 interrupt inputs for triggering OB3 as interrupt service routine.

Priorities

Table 12-1. Priorities of the Organization Blocks

Organization Block		Can be Interrupted by	Priority
OB2	Asynchronous interrupt processing		High ↓
OB3	Synchronous interrupt processing	OB2	
OB13	Time-controlled program scanning	OB2, OB3	
OB1	Cyclic program scanning	OB2, OB3, OB13	Low

Differences between OB2 and OB3 Interrupts

Table 12-2. Differences Between OB2 and OB3 Interrupts

	OB2 Interrupt	OB3 Interrupt
Hardware	4 special interrupt inputs (sub D connector)	16 onboard DIs and 4 special hardware interrupt inputs
Interrupt response time	3 ms typ. (also see Section 12.4)	8 to 14 ms typ. (also see Section 12.4)
Programming restrictions	Only a few command types possible	None

12.2 Asynchronous Interrupt Processing in OB2

The S5-95F is equipped with four separate failsafe interrupt inputs (I 59.0 to I 59.3) which trigger asynchronous interrupt processing when there is a change (falling edge) in a process signal. Asynchronous interrupt processing means that each subunit invokes OB2 as soon as it detects a change in the signal at an interrupt input, without regard to the other subunit's signals.

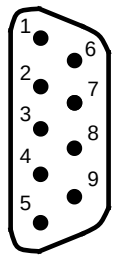
The two subunits invoke OB2 at different times (asynchronously). The main advantage of asynchronous interrupt handling is that an onboard output, and thus the relevant circuit, can be reset very quickly via the user program.

The following is required for the interrupt inputs:

- They must be wired via the sub D connector
- If necessary, the parameters for interrupt processing must be changed accordingly in DB1
- These parameters must then be evaluated in OB2.

Connector Pin Assignments for Counter and Interrupt Inputs

Table 12-3. Pin Assignments for the 9-Pin Sub D Connector

Sub D Connector	Pin Number Subunits A and B		Description
	1	1	L + (24 V DC supply)
	2	2	Counter B input (IW 38)
	3	3	Reference potential (M)
	4	4	Counter A input (IW 36)
	5	5	Reference potential (M)
	6	6	Interrupt input I 59.0
	7	7	Interrupt input I 59.1
	8	8	Interrupt input I 59.2
	9	9	Interrupt input I 59.3

Sensor Requirements

The permissible discrepancy time for interrupt inputs (OB2) is fixed at approx. 1 ms.
For interrupt inputs (OB2), use either

- Single-channel failsafe sensors
  or
- High-speed two-channel sensors with contacts so synchronous that identical circuit states are possible within 1 ms.

Interrupt Generation

An interrupt is triggered by a ***negative edge*** at an enabled interrupt input.

In the event of an interrupt, the S5-95F automatically invokes OB2. If OB2 has not been programmed, the cyclic or time-controlled program is immediately resumed.

The cyclic program can be interrupted after each STEP 5 statement.

Interrupt processing can be disabled with the IA operation and (re)enabled with the RA operation. RA is the default. When "IA" is in force, interrupts are stored (see Interrupt priority).

Interrupt Priority

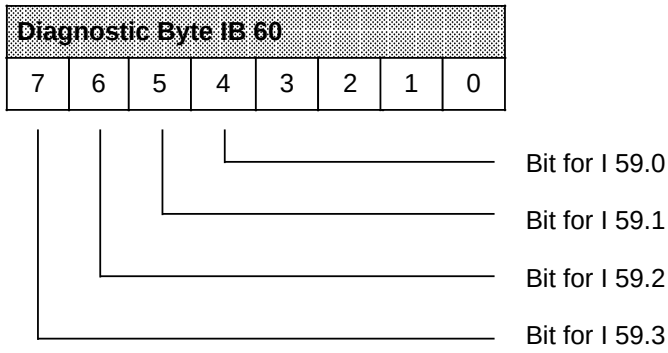
The OB2 interrupt processing routine cannot be interrupted. If other OB2 interrupts are generated while OB2 is executing, they are stored and OB2 reinvoked when it has terminated.

Ascertaining the Cause of an Interrupt

Negative signal edges generate an interrupt at one or more interrupt inputs.

When an interrupt is generated,

- OB2 is invoked, if programmed
- The relevant bit in diagnostic byte IB 60 is set to "1", even if OB2 has not been programmed. The bits are read immediately into the PII (there is no waiting for the cyclic read-in).



Other Responses to Passivation of Interrupt DI Byte 59

Interrupt DI and onboard DQ must be assigned to the same signal group if the process requires that passivation of interrupt DI byte 59 trigger immediate disabling of onboard DQ byte 32.

If you assign interrupt DI and onboard DQ to ***different signal groups*** on parameter assignment, the onboard DQ will ***not*** be ***disabled automatically*** on passivation of the interrupt DI. In this case, you must include the response for the onboard DQ in the user program.

12.2.1 Programming OB2

The primary objective of OB2 interrupt processing is the fastest possible resetting of an onboard DQ in response to changes at an interrupt input.

Operations Permissible in OB2

Table 12-4. Operations Permissible in OB2

Operation	Operand	Description/Comments
A AN O ON	I32.0 ... I33.7 I56.0 ... I56.7 I60.4 ... I60.7	Binary logic operation with onboard DI Binary logic operation with communication bits Binary logic operation with diagnostic bits
R	I60.4 ... I60.7 Q32.0 ... Q32.7	Resetting of diagnostic bits Resetting of onboard I/Os
A(O( )		Parenthesized expressions
BEU BEC BE		Block End identification

Note

The runtime for OB2 may not exceed 2 ms.

Communication and Diagnostic Bytes for Responses to OB2 Interrupts

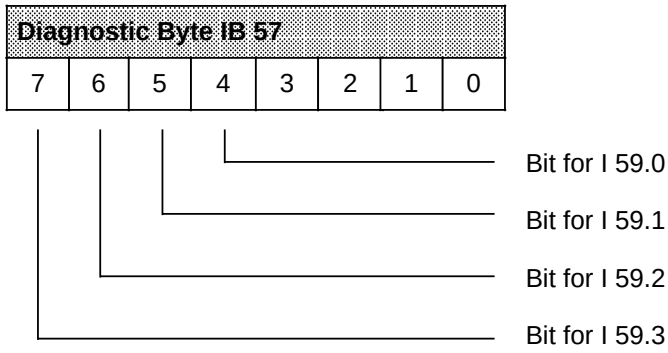
Input bytes IB 56 to IB 58 have been reserved in order to make it possible to respond to interrupts without invoking OB2.

Additional Communication and Diagnostic Bytes for Processing Interrupts with Routines Other than OB2

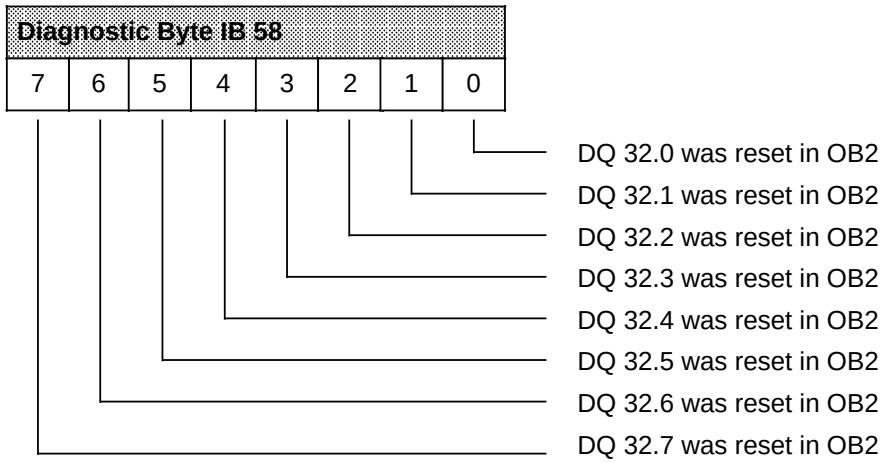
Table 12-5. Communication and Diagnostic Bytes for Responses to OB2 Interrupts

Address	Task	Comments
IB 56	Communication byte	The user program, except OB2, can write random information into IB 56. The user can then evaluate this information in OB2.
IB 57	Diagnostic byte for "OB2 interrupts"	The operating system uses IB 57 to identify OB2 interrupts. This byte can then be evaluated and reset in the user program (outside OB2).
IB 58	Diagnostic byte for "DQs reset in OB2"	The operating system uses IB 58 to identify the onboard DQs reset in OB2. This byte can then be evaluated and reset in the user program (outside OB2).

Diagnostic Byte IB 57



Diagnostic Byte IB 58



12.2.2 Programming Interrupt Responses in OB2

Interrupt-driven program scanning is possible only when the following prerequisites have been fulfilled:

- Interrupt inputs have been initialized in DB1
- The S5-95F must be in the "POWER ON" state and set to "RUN"
- Interrupt processing must not have been disabled with "IA" (see section 8.2.8)
- OB2 must be programmed, whereby heed must be paid to the restrictions listed in section 12.2.1.

Example for Programming OB2

A negative signal edge at interrupt input I 59.0 triggers an interrupt.

This, in turn,

- Invokes OB2, if programmed, and
- Sets bit 60.4 in diagnostic byte IB 60 to "1" if OB2 has not been programmed.

Table 12-6. Example for Programming OB2

Example	STL	Description
Outputs 32.0 and 32.1 are to be reset in response to an interrupt at bit 59.0.	OB2 A I 60.4 R I 60.4 R Q 32.0 R Q 32.1	Ascertain cause of INT Reset diagnostic bit When interrupt at I 59.0, reset outputs Q 32.0 and Q 32.1.
If interrupt occurred while OB2 was executing (at I 59.0), also reset Q 32.2 5 s later.	OB1 A I 57.4 R I 57.4 L KT 500.0 SS T 4 A T 4 R Q 32.2 R T 4	ID: Interrupt at I 59.0 Reset diagnostic bit Start delay timer Q 32.2 is reset after 5 s

Note

The diagnostic bits in IB 60 must be reset to "0" when the interrupt has been processed. The system resets the appropriate bits when the next interrupt is generated.

12.2.3 Connecting Interrupt Inputs

Example 1: A single-channel failsafe sensor S1 is to be connected to interrupt I 59.0 (type B).

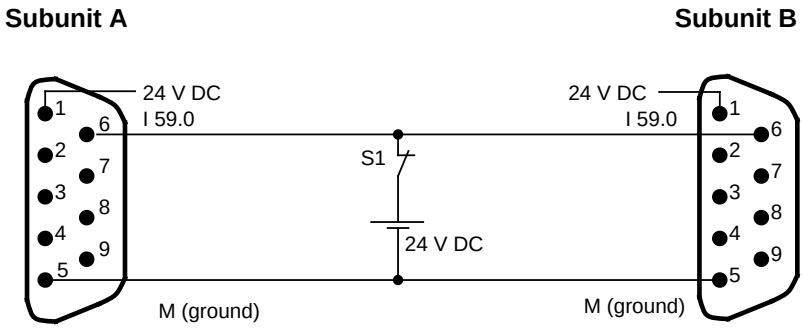


Figure 12-1. Connecting a Failsafe Sensor to an Interrupt Input

Example 2: A single-channel failsafe sensor S1, with line monitoring via DQ 33.1, is to be connected to interrupt input I 59.0.

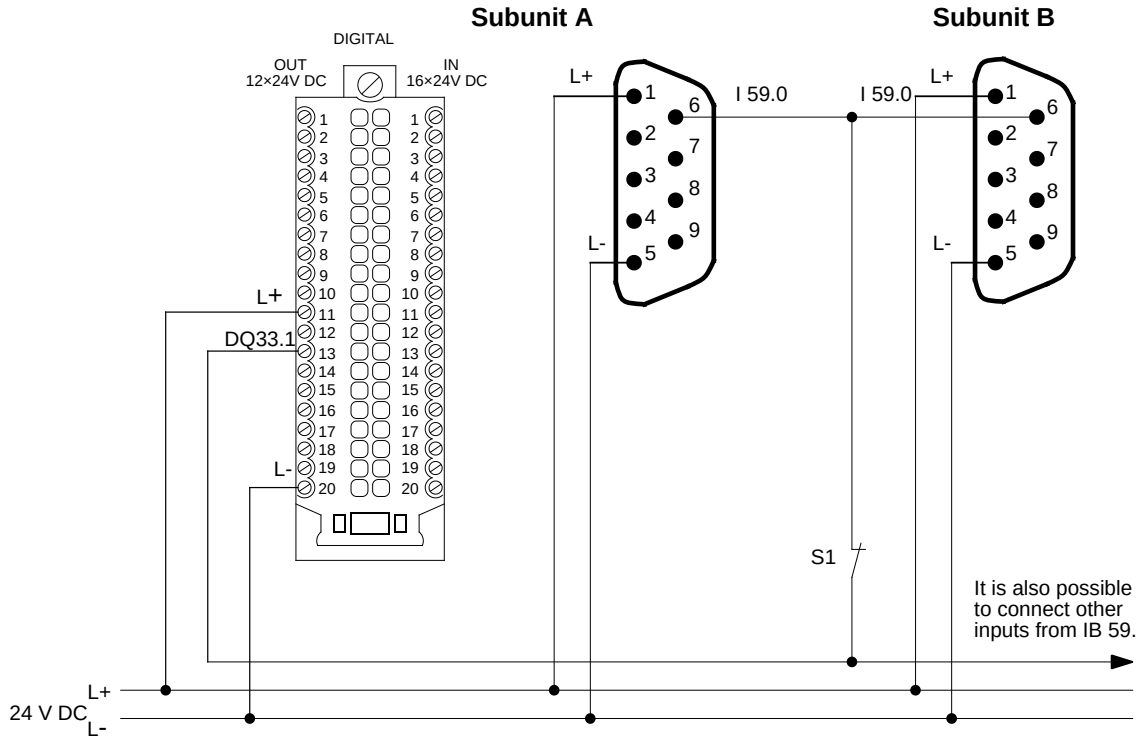


Figure 12-2. Connecting a Failsafe Sensor and a Line Monitor to an Interrupt Input

Example 3: Two sensors, S1a and S1b, are to be connected to interrupt input I 59.0 (type C).

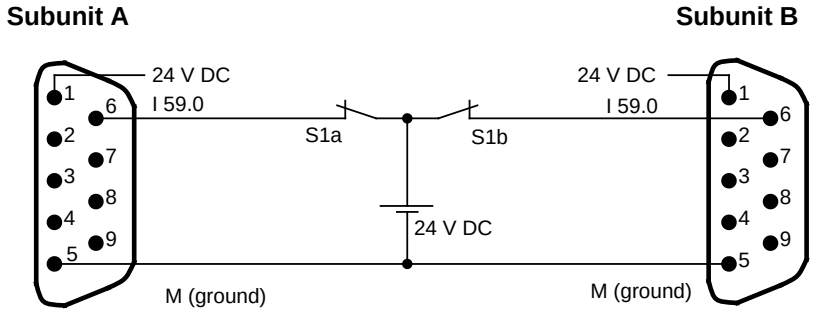


Figure 12-3. Connecting Two Independent Sensors to an Interrupt Input

Example 4: Two sensors, S1a and S1b, with line monitoring compared with other signal lines via test DQ 33.1, are to be connected to interrupt input I 59.0 (type E).

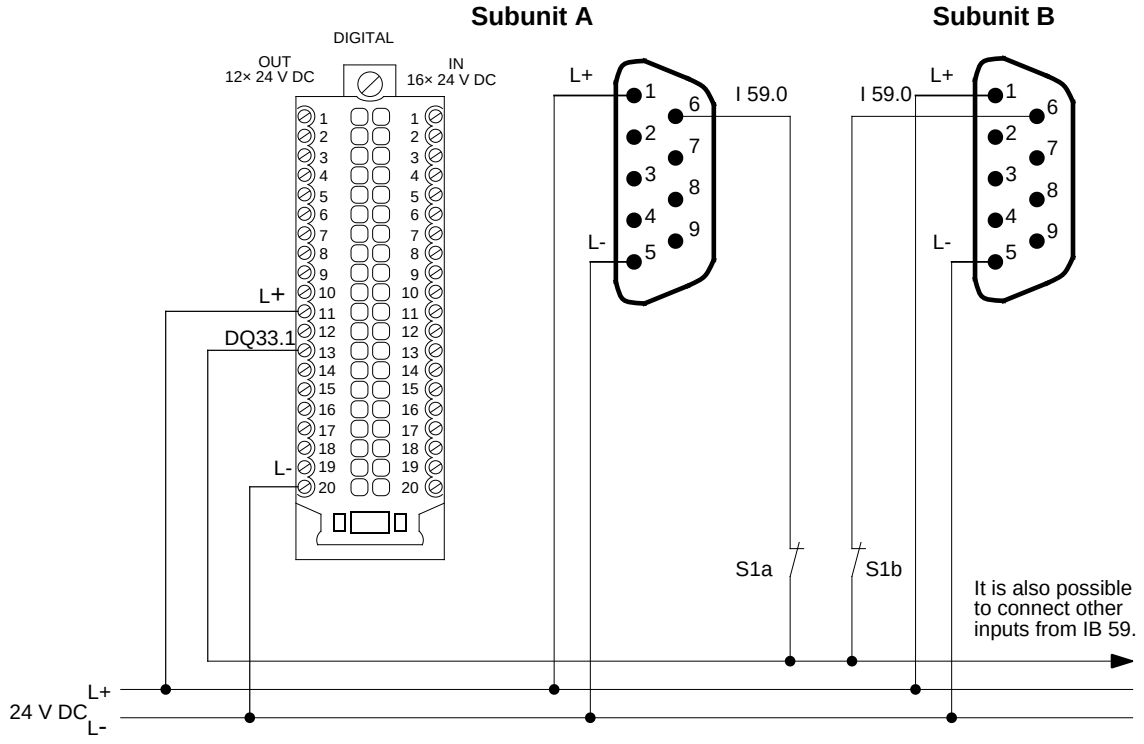


Figure 12-4. Connecting Two Independent Sensors and a Line Monitor to an Interrupt Input

Note

A separate test DQ bit is required for the DI line test. You may only use this test DQ bit for monitoring DIs from IB 59. DIs from other input bytes must not be connected to this test DQ bit.

12.3 Synchronous Interrupt Processing in OB3

Synchronous interrupt processing means that the signal state change must take place in both subunits before OB3 is invoked. Execution of OB3 takes place at the same time (i.e. is synchronous) in both subunits.

All onboard inputs (I 32.0 to I 33.7 and I 59.0 to I 59.3) may also be used as synchronous interrupt inputs. Those to be used as interrupt inputs must be initialized as such in DB1.

The interrupt inputs must be

- Connected via the 40-pin onboard connector (I 32.0 to I 33.7) and/or the sub D connector (I 59.0 to I 59.3 and counters A and B),
- Initialized, and thus enabled, in DB1 and
- Evaluated in OB3.

Connector Pin Assignments for Interrupt Inputs

See Connector Pin Assignments, section 4.7

Interrupt Triggering

The S5-95F executes interrupt processing routine OB3

- When an onboard counter reaches the specified comparison value
- On a) a falling signal edge or b) a rising signal edge or c) on a rising or falling signal edge at an OB3 interrupt input (the type of signal edge is specified in DB1 using the COM 95F software).

> **Safety Note**
>
> The choice of the interrupt-triggering signal edge is of consequence as regards safety. Should you decide on edge b) or c), you must prove its safety to the inspector at the time of the acceptance test.

If the appropriate parameters are assigned, the S5-95F responds to a signal edge change at an interrupt DI as follows:

- On the falling edge as soon as a signal change from 1 to 0 is detected ***in one subunit***.
- On the rising edge when a signal change from 0 to 1 is detected ***in both subunits***.

Special Feature of Interrupt DIs with OB1-Oriented Discrepancy Time

In systems with basic units 095-8FA02 and 095-8FB01 you can also use the interrupt DIs with an OB1-oriented discrepancy time.

To ensure detection of a continuous 1 fault by the S5-95F, the duration of the ”0” and ”1” signal states must be at least: ***parametrized discrepancy time + one OB1 cycle***.

Interrupt Priority

OB3 can interrupt the cyclic or time-controlled program after each STEP 5 statement. If OB3 has not been programmed, the cyclic or time-controlled program scan is resumed immediately after the interrupt has been triggered.

The OB3 interrupt processing routine can be interrupted by an OB2 interrupt, but not by another OB3 interrupt or an OB13 call. If OB3 interrupts are generated while the OB3 service routine is executing, they are stored so that OB3 can be reinvoked once it has terminated.

Interrupt processing can be disabled with IA and re-enabled with RA. The default is RA. Interrupts are stored while "IA" is in force (see Interrupt priority).

Note

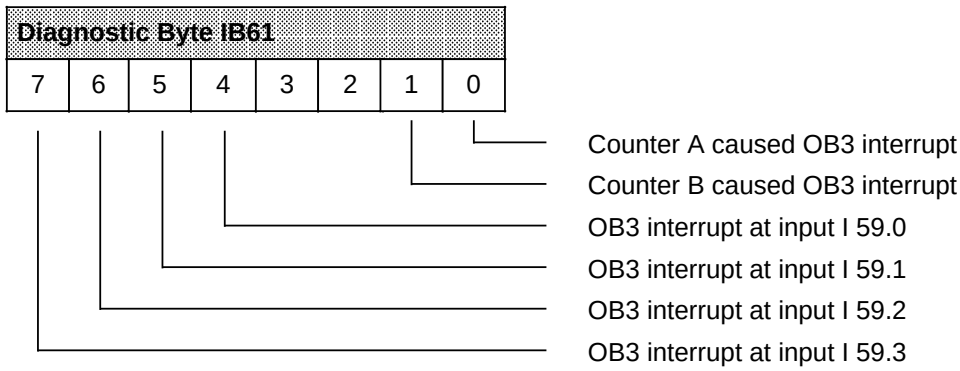
The block nesting depth of 16 may never be exceeded, even during interrupt processing.

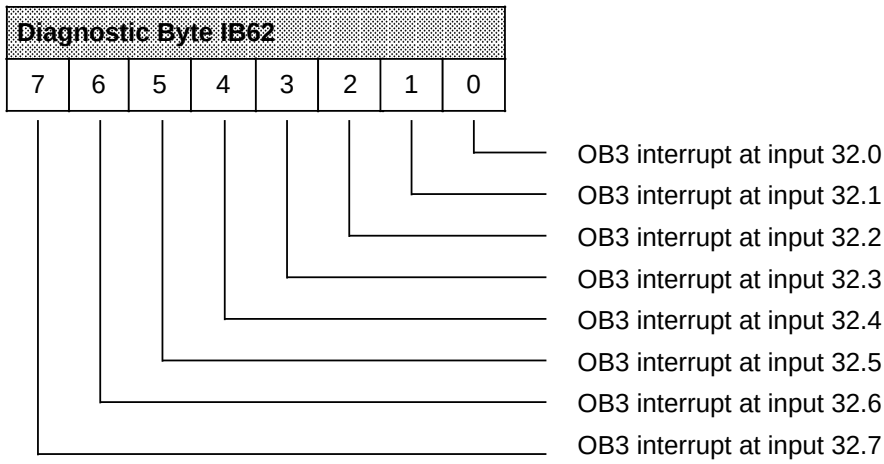
Ascertaining the Cause of an Interrupt

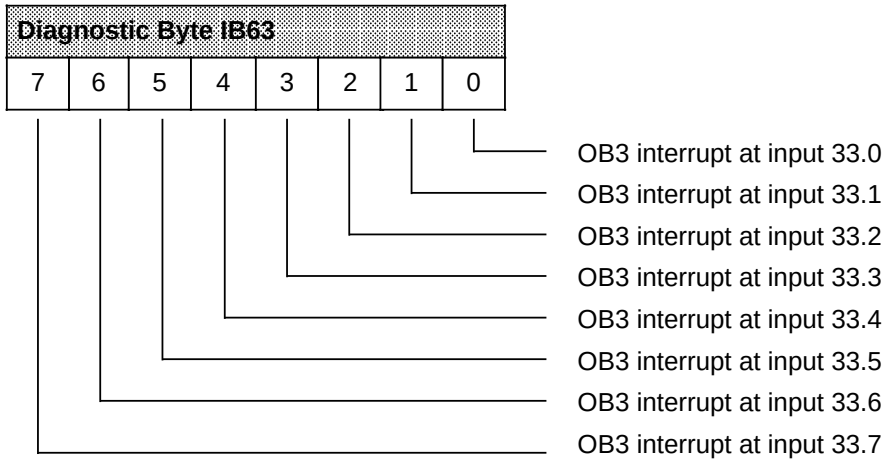
Positive and/or negative signal edges trigger interrupts at one or more interrupt inputs.

In this case,

- OB3 is invoked, if programmed
- The relevant bit in diagnostic byte IB 61 .. 63 is set to "1", even when OB3 has not been programmed.







The bits are immediately read into the PII (there is no waiting until it is time for the cyclic read-in).

12.3.1 Programming OB3

In order to be able to respond quickly to an interrupt, OB3 should be kept as short as possible. Be sure to optimize the OB3 interrupt processing routine carefully.

Note

- In OB3, direct access is allowed to onboard I/Os only.
- The OB3 runtime may not exceed 8 ms.

The T PY/PW statement transfers the data directly to the relevant onboard outputs, and to the "normal" PIQ. There is neither an interrupt PII nor an interrupt PIQ.

Note

When flags used in the cyclic program are also used in an interrupt processing routine, they must be saved before the service routine executes (e.g. in a data block) and reloaded in the cyclic program.

12.3.2 Programming Interrupt Responses in OB3

Interrupt-driven program scanning is possible only when the following prerequisites have been fulfilled:

- Interrupt inputs have been enabled in DB1.
- The programmable controller must be in the "POWER ON" state and set to "RUN".
- Interrupt processing must not have been disabled via an "IA" statement (see section 8.2.8).
- OB3 must have been programmed.

Example for Programming OB3

A negative edge at interrupt input I 33.5 triggers an interrupt.

When this occurs,

- OB3 is invoked, if programmed, and
- Bit 63.5 in IB 63 is set to "1", even if OB3 has not been programmed.

Table 12-7. Example for Scanning the Diagnostic Bytes to Determine the Cause of an Interrupt

Example	STL	Explanation
The OB3 interrupt processing routine in FB10 is to be executed in the event of an interrupt at I 33.5	OB3 A I 63.5 R I 63.5 JC FB 10 . . . BE	 Interrupt at I 33.5 Reset diagnostic bit I 63.5 Execute program (response to interrupt) in FB10

Note

The diagnostic bits in IB 61 ... IB 63 must be reset to "0" immediately following execution of the interrupt processing routine, at the latest. The system resets the bits when the next interrupt is generated.

12.4 Interrupt Response Times for the S5-95F

The interrupt response time is the time between an edge change at the terminals of an interrupt input and the appropriate signal change at the terminals of an output on the basic unit. Any delays attributable to the cables, sensors or actuators are not part of the interrupt response time.

The following section covers response times under worst-case conditions.

Interrupt Response Time for OB2 Interrupts

The interrupt response time for OB2 interrupts, i.e. $T_{OB2\ response}$, depends on:

- The S5-95F's internal delay for OB2 interrupts (max. 5.6 ms).
- The execution time T_{OB2} of organization block OB2 (should not exceed 2 ms, see section 12.2.1).

In the majority of applications, the programs in OB2 are relatively short and the execution time of OB2 is less than 0.2 ms. You can expect a ***typical interrupt response time*** of 3 ms for OB2 interrupts.

Typical Interrupt Response Time for OB2 Interrupts

If you have not disabled an interrupt (IA/RA operations) in the user program, the ***typical interrupt response time*** is calculated as follows:

$T_{OB2\ response,\ typical} = 2.8\ ms + T_{OB2}$

If you disable an interrupt in the user program, the interrupt response time will be extended by the execution time of the program sequence for which the interrupt is disabled (max. 5 ms).

Interrupt Response Time for OB2 Interrupts under Worst-Case Conditions

If you have not disabled an interrupt in the user program (IA/RA operations), the ***interrupt response time under worst-case conditions*** is calculated as follows:

$T_{OB2\ response,\ worst\ case}\ \ 5.6\ ms + T_{OB2}$

If you disable an interrupt in the user program, the interrupt response time will be extended by the execution time of the program sequence for which the interrupt is disabled (max. 5 ms).

Minimum Signal Duration for OB2 Interrupts

> **Note**
>
> An OB2 interrupt is triggered by a negative edge at an OB2 interrupt input. The S5-95F can detect the edge under worst-case conditions only when signal levels "0" and "1" have a duration of ***at least 4 ms***. For this reason, be sure to use suitable sensors (do not use fleeting contacts or rebound switches).

Frequency of OB2 Interrupts

> **Note**
>
> The primary objective of interrupt processing in OB2 is to reset an onboard DQ as quickly as possible when the signal edge at an interrupt input changes. An OB2 interrupt input is not suitable for counting high-frequency periodic pulses. As a guideline, we would recommend that OB2 ***not*** be invoked ***more often than every 50 ms*** via the same interrupt input. If it is invoked more frequently, the S5-95F might detect a discrepancy time error at the interrupt DI. In this case, the S5-95F reacts by issuing an error message and performing the parameterized system response (STOP or passivation).

Response Time for OB3 Interrupts

The response time for OB3 interrupts, $T_{OB3\ response}$, depends on:

- the internal S5-95F delay for OB3 interrupts (max. 12 ms when no OB2 interrupts are used and max. 16 ms when both OB2 and OB3 interrupts are used)
- the execution time T_{OB2} of organization block OB2 (should not exceed 2 ms; see section 12.2.1)
- the execution time T_{OB3} of organization block OB3 (should not exceed 8 ms; see section 12.3.1)
- the number of interrupt input bytes n_{bytes} used for interrupt processing with OB3

In the majority of applications, the programs in OB2 and OB3 are relatively short. The execution time of OB2 is generally less than 0.2 ms and that of OB3 less than 1 ms. You can therefore expect a ***typical interrupt response time*** of 8 to 14 ms for OB3 interrupts.

In individual cases, the response time for OB3 interrupts may increase as discussed below:

OB3 Interrupts Only, no OB2 Interrupts: Interrupt Response Time for OB3 Interrupts in Individual Cases

Under worst-case conditions, the response time for OB3 interrupts if no OB2 interrupts are used is

$T_{OB3\ response,\ wort\text{-}case}$ 12 ms+1.25×T_{OB3}+1.5 ms×n_{bytes} (n_{bytes}=1, 2 or 3)

OB2 and OB3 Interrupts: Interrupt Response Time for OB3 Interrupts in Individual Cases

Under worst-case conditions, the response time for OB3 interrupts when OB2 interrupts are also used is

$T_{OB3\ response,\ worst\text{-}case}$ 16 ms+5×T_{OB2}+1.25×T_{OB3}+1.5 ms×n_{bytes} (n_{bytes}=1, 2, 3)

Minimum Signal Duration for OB3 Interrupts

Note

An OB3 interrupt is triggered by an edge at an OB3 interrupt input or when an onboard counter reaches a specific comparison value.
The S5-95F detects the edge at an OB3 interrupt input under worst-case conditions when signal states "0" and "1" have a duration of ***at least (5 ms + $T_{OB3\ response}$)***; in the case of counters, signal levels "0" and "1" must have a duration of **at least 0.5 ms**. For this reason, be sure to use suitable sensors and optimize OB2 and OB3 for the shortest possible execution time.

13 Connecting the S5-95F to SINEC L1 and PROFIBUS

13.1 Options for Connecting the S5-95F to the SINEC L1 LAN 13- 1

13.2 Non-Failsafe Data Interchange over SINEC L1 . . . . . 13- 3
13.2.1 Initializing the S5-95F for Non-Failsafe Data Interchange 13- 4
13.2.2 Coordinating Non-Failsafe Data Interchange in the User Program . . 13- 6
13.2.3 Transmitting Non-Failsafe Data 13- 7
13.2.4 Receiving Non-Failsafe Data 13- 8

13.3 Failsafe Data Interchange over SINEC L1 . . . . . 13- 10
13.3.1 Initializing the S5-95F for Failsafe Data Interchange 13- 12
13.3.2 Coordinating Failsafe Data Interchange in the User Program 13- 15
13.3.3 Sending Failsafe Data 13- 18
13.3.4 Receiving Failsafe Data 13- 20

13.4 SINEC L1 Safety Times 13- 25
13.4.1 SINEC L1 Safety Time for Receive 13- 25
13.4.2 SINEC L1 Safety Time for Send 13- 28
13.4.3 Load Placed on the System by the SINEC L1 Channel 13- 29
13.4.4 Response Time During SINEC L1 Traffic 13- 30
13.4.5 Transmission of Error Messages to the SINEC L1 Master 13- 31
13.4.6 Example: SINEC L1 Cycle Time and SINEC L1 Safety Time 13- 31

13.5 Connecting the S5-95F to the PROFIBUS 13- 34

Figures

13-1 Subunit A on a Single SINEC L1 Channel 13- 1
13-2 Subunit B on a Single SINEC L1 Channel 13- 2
13-3 Subunits on Cable-Redundant, Double SINEC L1 Channel, Bus Masters in One CC 13- 2
13-4 Subunits on Cable-Redundant, Double SINEC L1 Channel, Bus Masters in Different CCs 13- 2
13-5 Data Interchange Between Transmitter and Receiver (Principle) 13- 6
13-6 Structure of the Send Mailbox 13- 7
13-7 Structure of the Coordination Byte Send (KBS) 13- 7
13-8 Structure of the Receive Mailbox 13- 8
13-9 Structure of the Coordination Byte for Receive (KBE) 13- 9
13-10 Data Interchange Between Transmitter and Receiver (Principle) 13- 15
13-11 Network structure 13- 31
13-12 Subunit A via CP 541 to the PROFIBUS 13- 34

Tables

13-1 Characteristics of Non-Failsafe Data Interchange 13- 3
13-2 Initializing the Interface for Non-Failsafe Data Interchange 13- 5
13-3 Overview of Message Modes for the S5-95F 13- 10
13-4 Characteristics of Failsafe Data Interchange 13- 11
13-5 Initializing the Interface for Failsafe Data Interchange 13- 13

13 Connecting the S5-95F to SINEC L1 and PROFIBUS

SINEC L1 is a local area network (also sometimes called a bus, or channel) for interconnecting SIMATIC S5 programmable controllers; it functions on the master-slave principle.

More detailed information on SINEC L1 local area networks can be found in the "SINEC L1 Manual". It has been assumed that the reader of this document is familiar with how a SINEC L1 LAN works.

The S5-95F can be used as slave station. The required information is presented in the following manual sections.

13.1 Options for Connecting the S5-95F to the SINEC L1 LAN

There are two ways to interface an S5-95F to SINEC L1:

- For a pure point-to-point connection between master and slave, the programmable controllers are interfaced via a direct cable connection (possible only when the two units are less than 100 m apart). For a point-to-point connection, use a four-wire shielded cable with a cross section of no less than 0.14 mm^2. It is recommended that SIMATIC cable 6ES5 707-1AA00 be used. If you have any questions, please contact your local Siemens branch office.
- If there are several nodes on the channel, connect the controllers using BT 777 bus terminals. A bus terminal serves as level converter.

The hardware configurations for failsafe and nonfailsafe data transfer are identical. A separate power supply for BT 777s, such as that required for the S5-115F, is unnecessary.

> **Warning**
>
> To ensure protection against dangerous touch voltage, all SINEC L1 components must be operated by electrically separate power supplies.

Subunit A on a Single SINEC L1 Channel

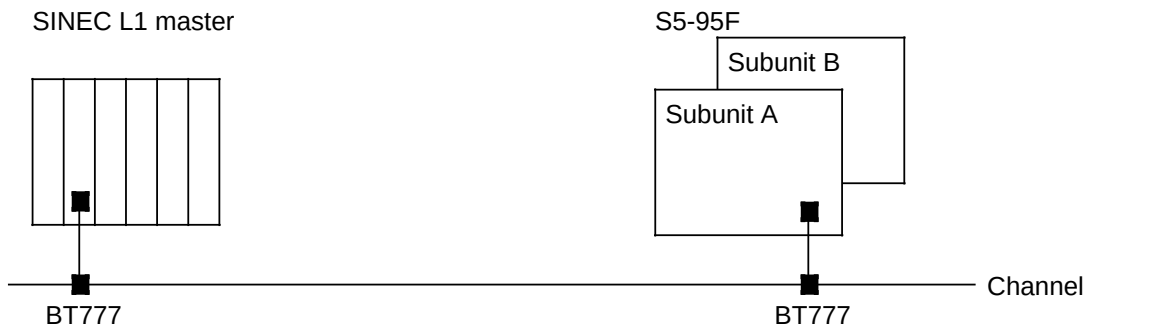


Figure 13-1. Subunit A on a Single SINEC L1 Channel

Subunit B on a Single SINEC L1 Channel

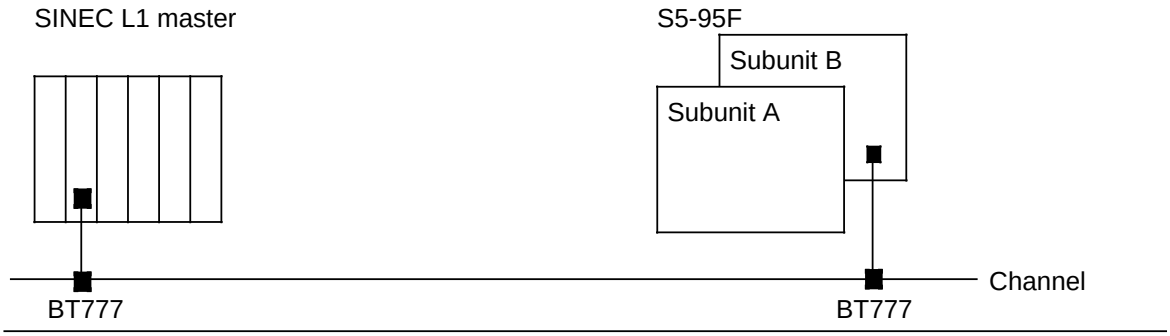


Figure 13-2. Subunit B on a Single SINEC L1 Channel

Both Subunits on a Cable-Redundant, Double SINEC L1 Channel

When a cable-redundant, double SINEC L1 channel is used, the configurations shown in Figures 13-3 and 13-4 can

- Send and receive on both channels, thus enabling high-availability interchange
- Send data on one channel and receive data on the other
- Distribute data traffic between the two channels as needed for maximum expediency.

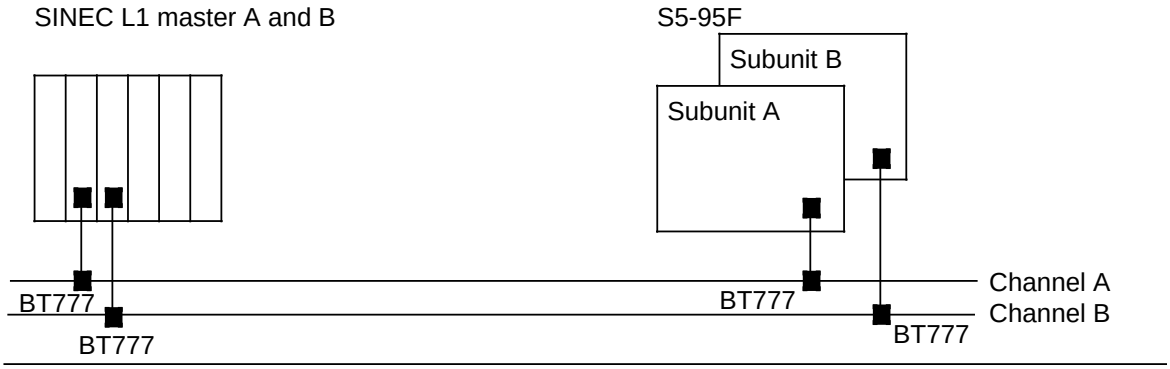


Figure 13-3. Subunits on a Cable-Redundant, Double SINEC L1 Channel, Bus Masters in One CC

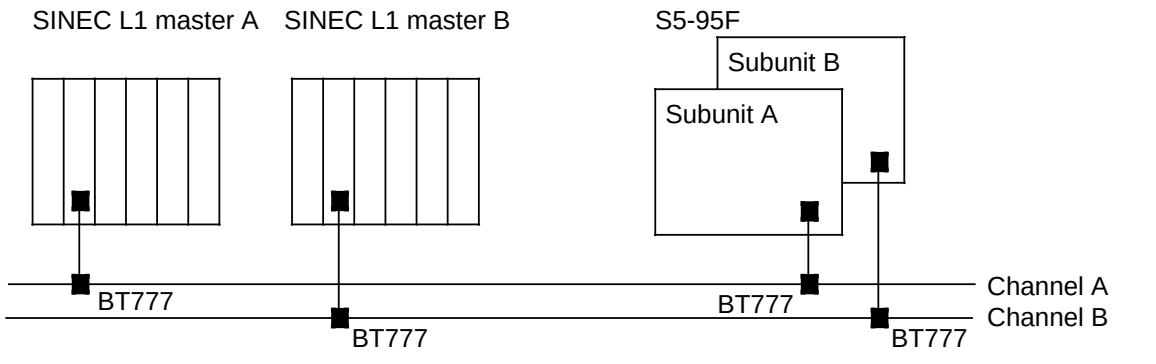


Figure 13-4. Subunits on a Cable-Redundant, Double SINEC L1 Channel, Bus Masters in Different CCs

Note

Two CP 530s as SINEC L1 bus masters are required when using the double SINEC L1 bus. Availability can be increased by plugging the two CP 530s into two different central controllers.

13.2 Non-Failsafe Data Interchange over SINEC L1

Non-failsafe data interchange is possible with all SINEC L1 nodes, and is the same as the standard data interchange method used by all SIMATIC S5 U-range controllers.

Characteristics of Non-Failsafe Data Interchange over SINEC L1

Table 13-1. Characteristics of Non-Failsafe Data Interchange

Characteristics of Non-failsafe Data Interchange	
Number of slaves with which the S5-95F can communicate:	Max. 30
Permissible frame length:	Max. 64 bytes
Reaction-free connection:	Yes
Failsafe frames possible:	No
Broadcast frames possible:	Yes
Interrupt frames possible:	No
Cable-redundant, high-availability SINEC L1 bus capability:	Yes
Location of Send and Receive mailboxes specifiable:	Yes

13.2.1 Initializing the S5-95F for Non-Failsafe Data Interchange

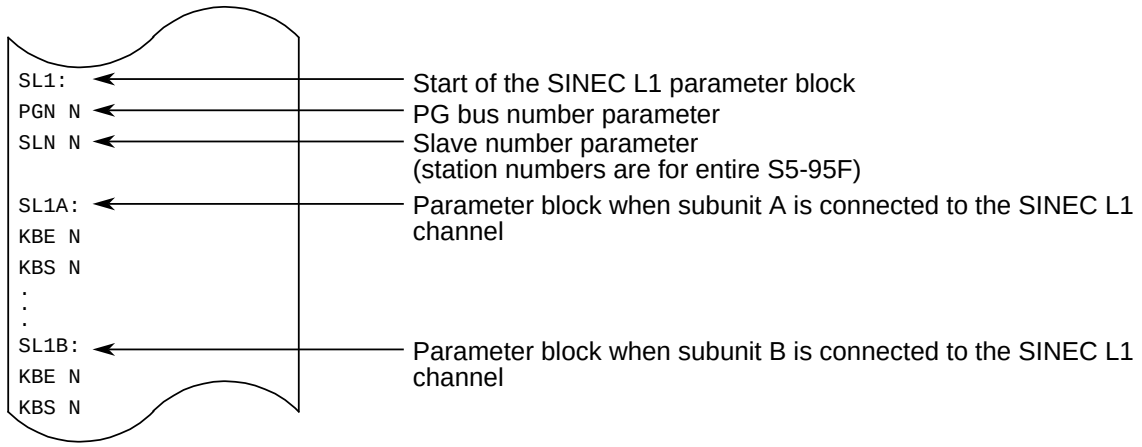
The S5-95F requires the following information in order to interchange data over the L1 channel:

- The slave number of the S5-95F
- The location of the Send data (data block or flag area)
  Designation: **Send Mailbox**, abbreviated **SF**
- Location of the Receive data (data block or flag area)
  Designation: **Receive Mailbox**, abbreviated **EF**
- Storage address for coordination info for transmitting data (e.g. "Send mailbox enabled")
  Designation: **Coordination Byte Send**, abbreviated **KBS**
- Storage address for coordination info for receiving data (e.g. "Receive data available")
  Designation: **Coordination Byte Receive**, abbreviated **KBE** and
- The programmer bus number assigned to the S5-95F (if programmer functions are to "execute" over the L1 channel)

This information must either be stored in DB1 or entered using the COM 95F software (refer to the COM 95F manual).

Proceed as follows to enter the information in DB1:

A default DB1 is integrated in the S5-95F's operating system; among other things, this data block provides parameter default values for transferring data over SINEC L1.
Load the default DB1 into your programmer (transfer function, source: PLC, destination: FD (PG))
Find the SINEC L1 parameter block; the block name for the PG/OP/SINEC L1 interface socket is "SL1:".
Note: Use capital letters only!



Edit the defaults as required; do not change the syntax.
You will find an example on the next page.
Transfer the edited DB1 to the PLC; the default DB1 is overwritten with the edited data block.

The new parameters do not go into force until the programmable controller is switched from STOP to RUN or from POWER OFF to POWER ON.

Example: The S5-95F is to be a slave with the slave number 2 on the SINEC L1 bus. Subunit A is interfaced to the L1 bus:

- The Send mailbox begins at data word 0 in DB2
- The Receive mailbox begins at data word 10 in DB2
- The Coordination Byte Send is flag byte 0
- The Coordination Byte Receive is flag byte 2
- The PG bus number is 1

Table 13-2 shows how to edit the default parameters for the example, and which parameter values are permissible:

Table 13-2. Initializing the Interface for Non-Failsafe Data Interchange

Parameter in Default DB1	Description	Value for Example	Permissible Values
...			
SL1:	Block ID: "SINEC L1"	—	—
PGN N	Programmer bus number (required to execute programmer functions over the L1 bus) Default: No	PGN 1	PGN x (x=1 to 30)
SLN N	Slave number of the S5-95F Default: No	SLN 2	SLN x (x=1 to 30)
SL1A:	Block ID: "Subunit A on SINEC L1"		
SF N	Location of the Send mailbox Default: No	SF DB2DW0	SF DBxDWy (X=2 to 251; y=0 to 255) or SF FYz (z=0 to 255)
EF N	Location of the Receive mailbox Default: No	EF DB2DW10	EF DBxDWy (X=2 to 251; y=0 to 255) or EF FYz (z=0 to 255)
KBE N	Location of the "Coordination Byte Receive" Default: No	KBE FY2	KBE MBx (x=0 to 255) or KBE DByDWz* (y=2 to 251; z=0 to 255)
KBS N	Location of the "Coordination Byte Send" Default: No	KBS FY0	KBS MBx (x=0 to 255) or KBS DByDWz* (y=2 to 251; z=0 to 255)

* The KBE/KBS is in the high-order byte of the specified data word.

13.2.2 Coordinating Non-Failsafe Data Interchange in the User Program

After initializing the parameters, you have to write the user program for interchanging data. The user program is dependent on the coordination information which the operating system makes available in the coordination bytes (see Figure 13-5).

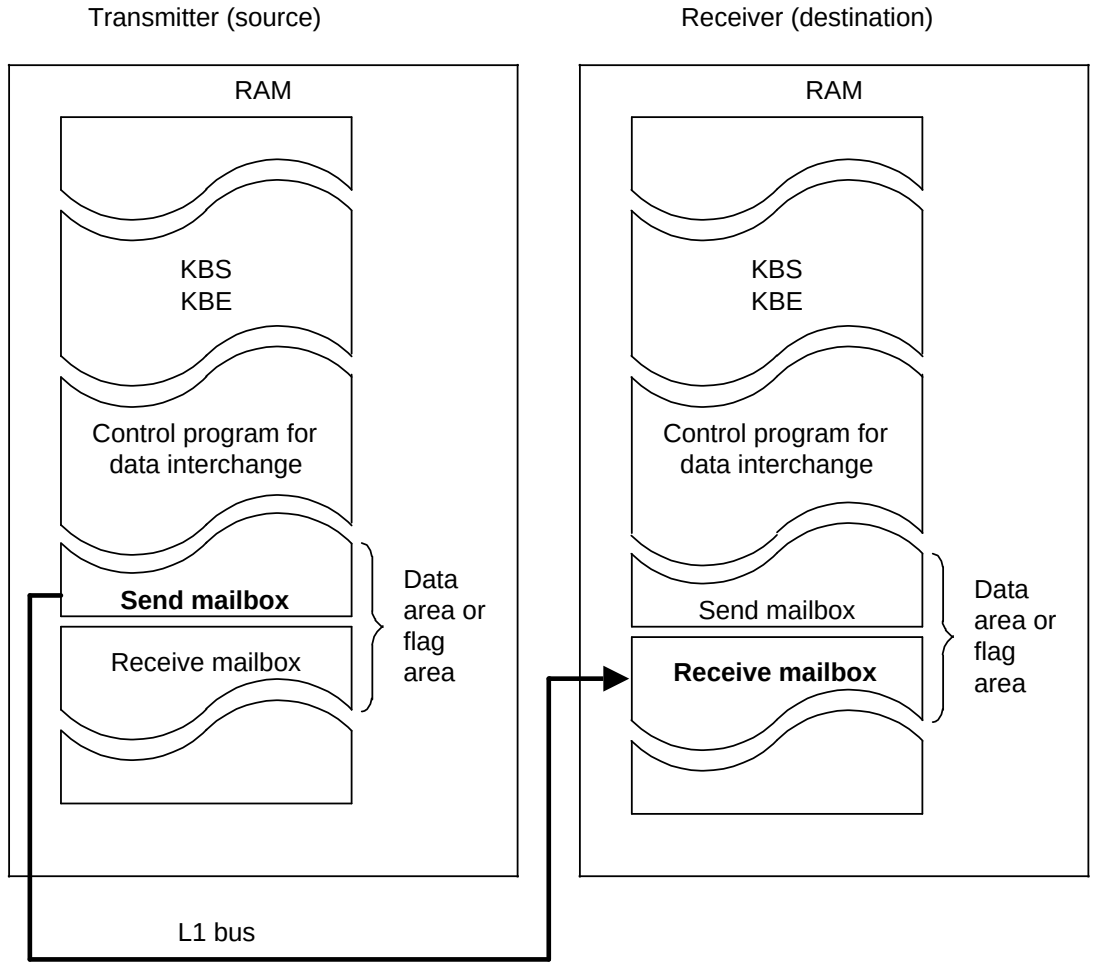


Figure 13-5. Data Interchange Between Transmitter and Receiver (Principle)

In the following paragraphs you will learn how to control the transmitting and receiving of data.

13.2.3 Transmitting Non-Failsafe Data

The prerequisites for transmitting data are as follows:

- The location of the Send mailbox has been specified in DB1 (see section 13.2.1)
- The data to be transmitted, additional information (such as the length of the Send data ("net data") and the destination slave number) have been forwarded to the Send mailbox.

Figure 13-6 shows which information has to be stored in what part of the Send mailbox.

Example:
Send mailbox in **flag area**
(starting with FY 1)

Flag byte 1	Length of the "net data" (in bytes (0 to 64))
Flag byte 2	Destination slave number*
Flag byte 3	Data bytes ("net data") max. 64 bytes
Flag byte 66	

Example:
Send mailbox in **data block**
(starting with DW 1)

	DL	DR
DW 1	Length of the "net data"	Destination slave number*
DW 2	1st data byte	2nd data byte
DW 33	63rd data byte	64th data byte

* Number of the receiver; 0 = Master
1 to 30 = Slaves
31 = Broadcast

Figure 13-6. Structure of the Send Mailbox

Structure of the Coordination Byte Send (KBS)

Figure 13-7 shows the structure of the Coordination Byte Send (KBS).

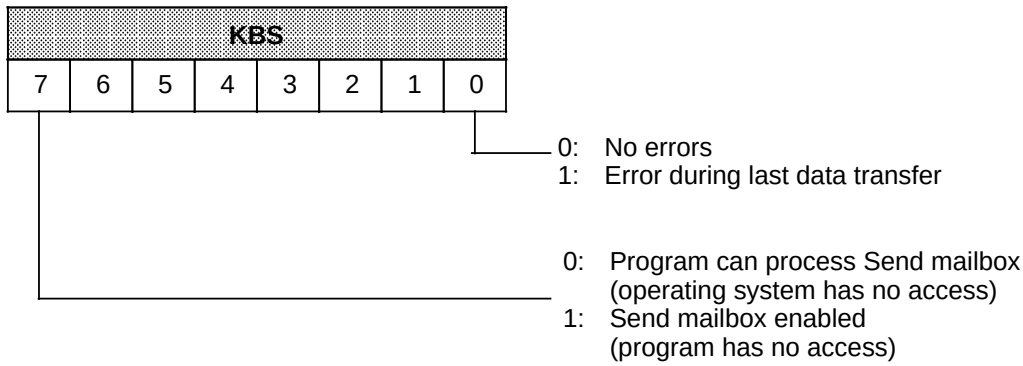


Figure 13.7 Structure of the Coordination Byte Send (KBS)

The user program for transmitting data should be structured as follows:

Check bit 7 in the KBS to see if data is currently being transferred
(As long as the PLC is transferring data, bit 7 remains set in the KBS. During this phase, never attempt to change the contents of the Send Mailbox or start another data transfer).

When bit 7 in the KBS has been reset:
Start the transfer by setting bit 7 in the KBS.

When the operating system has reset bit 7 following completion of the transfer: Check for errors.

If an error occurs during transfer, the operating system sets bit 0 in the KBS. The error, however, cannot be evaluated until KBS bit 7 has been reset.

13.2.4 Receiving Non-Failsafe Data

Safety Note

Regardless of the intended frame length, the S5-95F's ***Receive Mailbox for non-failsafe data interchange*** must
- either have a length of 66 bytes or
- be located at the end of the flag area or data block.

Prerequisites for receiving data:

The parameters for the locations of Receive Mailbox and Coordination Byte Receive (KBE) have been initialized in DB1 (see section 13.2.1). Figure 13-8 shows which information is stored where in the Receive Mailbox:

Example:
Receive Mailbox in **flag area**
(starting with flag byte 1)

FY 1	Length of the "net data" (in bytes)
FY 2	Source slave number*
FY 3	Data ("net data")
FY 66	

Example:
Receive Mailbox in **data block**
(starting with data word 1)

	DL	DR
DW 1	Length of the "net data"	Source slave number*
DW 2	1st data byte	2nd data byte
DW 3	3rd data byte	4th data byte
DW 33	63rd data byte	64th data byte

* Number of the sender; 0 =Master
1 to 30 =Slave

Figure 13-8. Structure of the Receive Mailbox

Structure of the Coordination Byte Receive (KBE)

Figure 13-9 shows the structure of the Coordination Byte Receive (KBE).

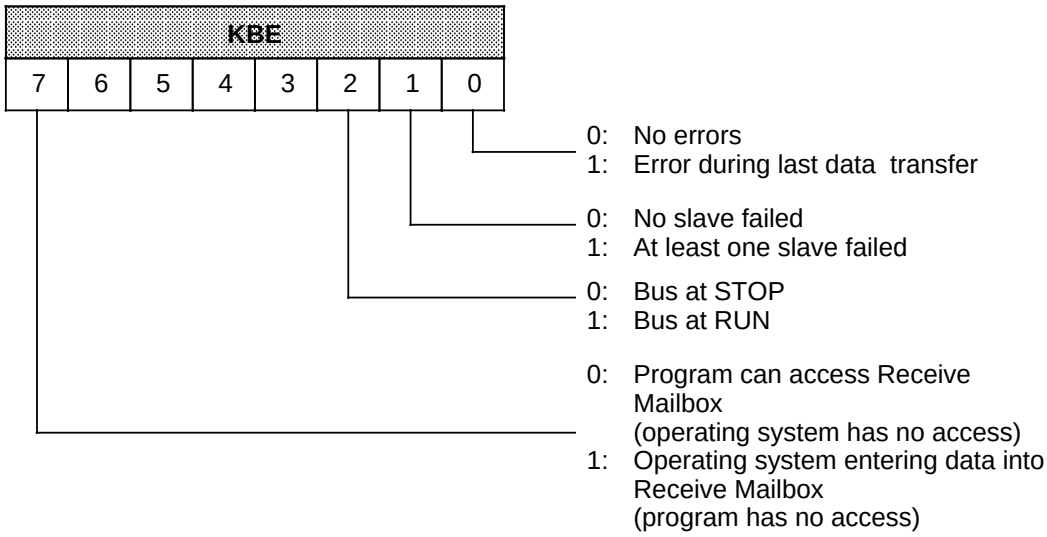


Figure 13-9. Structure of the Coordination Byte Receive (KBE)

Structure of the user program for receiving data:

Check bit 7 in the KBE to see whether you can retrieve data from the Receive Mailbox. In order for you to be able to read data out of the KBE, bit 7 must be "0".

The following errors and status codes are also flagged in the KBE:

- At least one slave failed
- Bus at RUN (STOP)

Special Features

If you reserved too little space in memory for the Receive mailbox, the available space is padded (flag area up to flag byte 255, data block up to and including the last data word); the remaining Receive data cannot be stored. In this case, the PLC does not report an overflow.

Sample programs for sending and receiving data can be found in the SINEC L1 Manual (in the section entitled "Programming").

Interval for Receiving Frames

> **Note**
>
> Please note that the S5-95F cannot receive a new frame until it has processed the previous one. You should therefore arrange communications between nodes so that frames are received at least 100 ms apart, otherwise the S5-95F may not accept the next frame.

13.3 Failsafe Data Interchange over SINEC L1

Failsafe data interchange is possible with SINEC L1 S5-95F and S5-115F slave nodes.

Failsafe data interchange is only possible when both sender and receiver are operating in the same message mode. Systems with basic units 095-8FA02 support the two message modes of the S5-115F in addition to the standard mode for the S5-95F

The following table shows the features of the three message modes.

Table 13-3. Overview of Message Modes for the S5-95F

Message mode	Features	Typical applications
95F	High-level protection of the destination slave No. Change byte for detection of new frame Depassivation of data paths Useful frame monitoring Nonfailsafe handshake	Standard mode for communication with S5-95F devices
115F-14	High-level protection of the destination slave No. No change byte for detection of new frame Automatic depassivation of data paths No useful frame monitoring No handshake	Communication with the S5-115F with CPU 942-7UF14 (without use of a CP 541)
115F-15	High-level protection of the destination slave No. Change byte for detection of new frame Automatic depassivation of data paths No useful frame monitoring No handshake	Communication with the S5-115F with CPU 942-7UF15 and/or connection of the S5-95F to a different bus system via CP541 Transmission of failsafe broadcast messages

Data Transmissions from the SINEC L1 Master

To ensure problem-free data traffic on the SINEC L1, failsafe data should always be transmitted more frequently than non-failsafe data.

For this reason, make sure that the SINEC L1 master does not transmit too often. Organize the SINEC L1 master's data transmissions, for instance, in the time-controlled program (e.g. OB13). The call interval should, if possible, not exceed the SINEC L1 safety time for transmitting (see section 13.4.2).

Characteristics of Failsafe Data Interchange over SINEC L1

Table 13-4. Characteristics of Failsafe Data Interchange

Characteristics of Failsafe Data Interchange	
Number of slaves to which the S5-95F can transmit:	2 (data paths 1 and 2)
Number of slaves from which the S5-95F can receive:	2 (data paths 1 and 2)
Permissible frame length (net data)	Max. 60 bytes
Reaction-free connection:	Yes
Failsafe frames possible:	Yes
Broadcast frames possible:	Yes, in 115F-15 mode
Interrupt frames possible:	No
Cable-redundant, high-availability SINEC L1 bus capability:	Yes

For security reasons, the S5-95F only allows either the transmission of failsafe broadcast frames or node-to-node transmission of failsafe frames via SINEC L1. These options are mutually interlocked by COM 95F.

Conditions for Failsafe Input/Output Signals

Safety Note

If failsafe input and output signals are to be transmitted over the SINEC L1 network, a "0" signal must always lead to a safe system state (***closed-circuit principle***). This is absolutely necessary, as the S5-95F erases the contents of the Receive Mailboxes in the event of a data transfer error.

Inputs are in a safe state when the transmitted "0" signal brings the process to the safe quiescent state. The input for an EMERGENCY STOP, for instance, must have a "1" signal during operation and be activated by a "0" signal.

Outputs are in a safe state when the transmitted "0" signal resets the output and brings the actuator connected to it to a safe state.

Special Features of Failsafe Data Interchange over SINEC L1

Note

It is not necessary to transmit and receive at least one useful frame within the SINEC L1 safety time. If necessary, the operating system sends test frames, unbeknownst to the user, in order to monitor the bus.

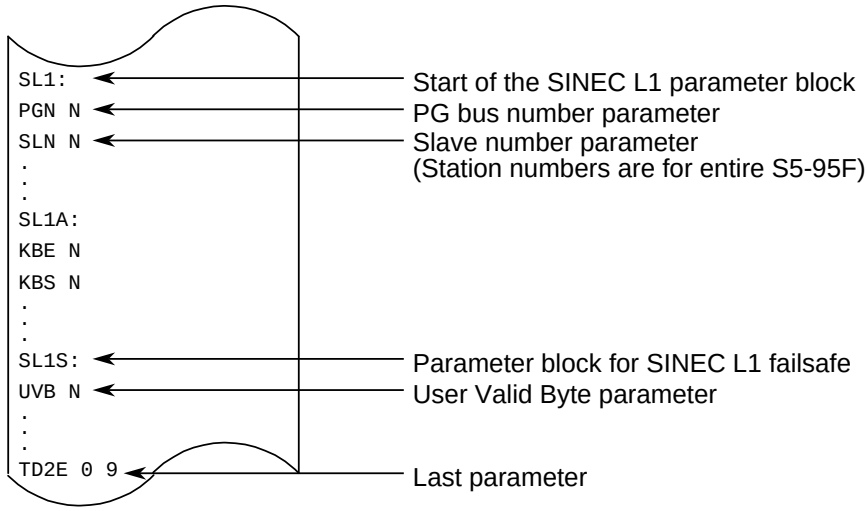
13.3.1 Initializing the S5-95F for Failsafe Data Interchange

The S5-95F must be supplied with the following information in DB1 in order to carry out failsafe data interchange over the L1 bus.

- The slave number assigned to the S5-95F
- Where to store the coordination info for transmitting and receiving
  Designation: User Valid Byte, abbreviated **UVB**
- The subunit interfaced to the SINEC L1 channel
- The slaves to be used for failsafe data transfer, and the data transfer mode
- Permissible SINEC L1 safety times
- How the S5-95F is to respond when it receives a bad frame

This information must either be stored directly in DB1 or the appropriate parameters initialized using the COM 95F software:

A default DB1 is integrated in the S5-95F's operating system, and contains, among other things, default parameters for data interchange over the L1 bus.
Load the default DB1 into your programmer (Transfer function, Source:PLC, Destination:FD (PG)).
Find the SINEC L1 parameter block; the block ID for the PG/OP/SINEC L1 interface socket is "SL1:".



Edit the default parameters to meet your needs; do not change the syntax. You will find an example on the next page.
Transfer the edited DB1 to the PLC; the default DB1 is overwritten.

The programmable controller does not interpret the edited parameters until the controller is switched from STOP to RUN mode or from POWER OFF to POWER ON.

Example: The S5-95F is to be a slave station and have the slave number 2 on the SINEC L1 bus. Subunit A is interfaced to the L1 bus.

The following conditions apply to data transfer:

- Coordination byte UVB is flag byte 10
- Data is to be transmitted over data path 1 to slave 13 and 14
- Data is to be received over data paths 1 and 2 from slaves 13 and 14
- Slave 13 is an S5-115F programmable controller with CPU 942-7UF15 and slave 14 is an S5-95F programmable controller
- The SINEC L1 safety time is 3 seconds for all connections
- If no valid frame can be received within the safety time, the S5-95F goes to STOP

Table 13-5. Initializing the Interface for Failsafe Data Interchange

Parameter in the Default DB	Description	Value for the Example	Permissible Values
SL1:	Block ID "SINEC L1"	—	—
PGN N	PG bus number (required for executing programmer functions over the L1 bus; default value is NO)	PGN 1	PGN x (x=1 to 30)
SLN N	Slave number Default value is NO	SLN 2	SLN x (x=1 to 30)
SL1S:	Block ID: "SINEC L1 failsafe"	—	—
UVB N	User Valid Byte	UVB MB 10	FY x (x=0 to 255)
D1S N	Data path 1 (send) is physically interfaced to subunit ...; default value is NO.	D1S A	D1S k k={N, A, B, H} where N=^no A=^subunit A B=^subunit B H=^subunits A and B (high-availability SINEC L1 bus)
D1E N	Data path 1 (receive) is physically interfaced to subunit ...; default value is NO.	D1E A	D1E k
D2S N	Data path 2 (send) is physically interfaced to subunit ...; default value is NO.	D2S A	D2S k
D2E N	Data path 2 (receive) is physically interfaced to subunit ...; default value is NO.	D2E A	D2E k

Table 13-5. Initializing the Interface for Failsafe Data Interchange (continued)

Parameters in Default DB	Description	Value for Example	Permissible Values
`SNTS1 0 9`	Slave no. and slave type to which data is to be transmitted over data path 1; default is no data traffic	`SNTS1 13 5`	`SNTS1 n t` n=1 to 31 0=No data traffic 1 ... 30=Slave No. 31=Broadcast t={1, 5, 9} where 1=115F-14 mode 5=115F-15 mode 9=95F mode
`SNTS2 0 9`	Slave no. and slave type to which data is to be transmitted over data path 2; default is no data traffic	`SNTS2 14 9`	`SNTS2 n t` n=Slave No. (1...30) t=Mode (1, 5, 9)
`SNTE1 0 9`	Slave no. and slave type from which data is to be received over data path 1; default is no data traffic	`SNTE1 13 5`	`SNTE1 n t` n=Slave No. (1...30) t=Mode (1, 5, 9)
`SNTE2 0 9`	Slave no. and slave type from which data is to be received over data path 2; default is no data traffic	`SNTE2 14 9`	`SNTS2 n t` n=Slave No. (1...30) t=Mode (1, 5, 9)
`TD1S 0`	Send safety time for data path 1	`TD1S 30`	`TD1S u` u=0, 3 to 1638 (times 100 ms)
`TD2S 0`	Send safety time for data path 2	`TD2S 30`	`TD2S u`
`TD1E 0 S`	Receive safety time for data path 1 and PLC response	`TD1E 30 S`	`TD1E u p` p={S, P} where S=STOP P=Passivation of the data paths with response on user level
`TD2E 0 S`	Receive safety time for data path 2 and PLC response	`TD2E 30 S`	`TD2E u p`

13.3.2 Coordinating Failsafe Data Interchange in the User Program

Once all parameters have been initialized, the user program for data interchange must be written. This program must access coordination info in the User Valid Byte (refer to Figure 13-10).

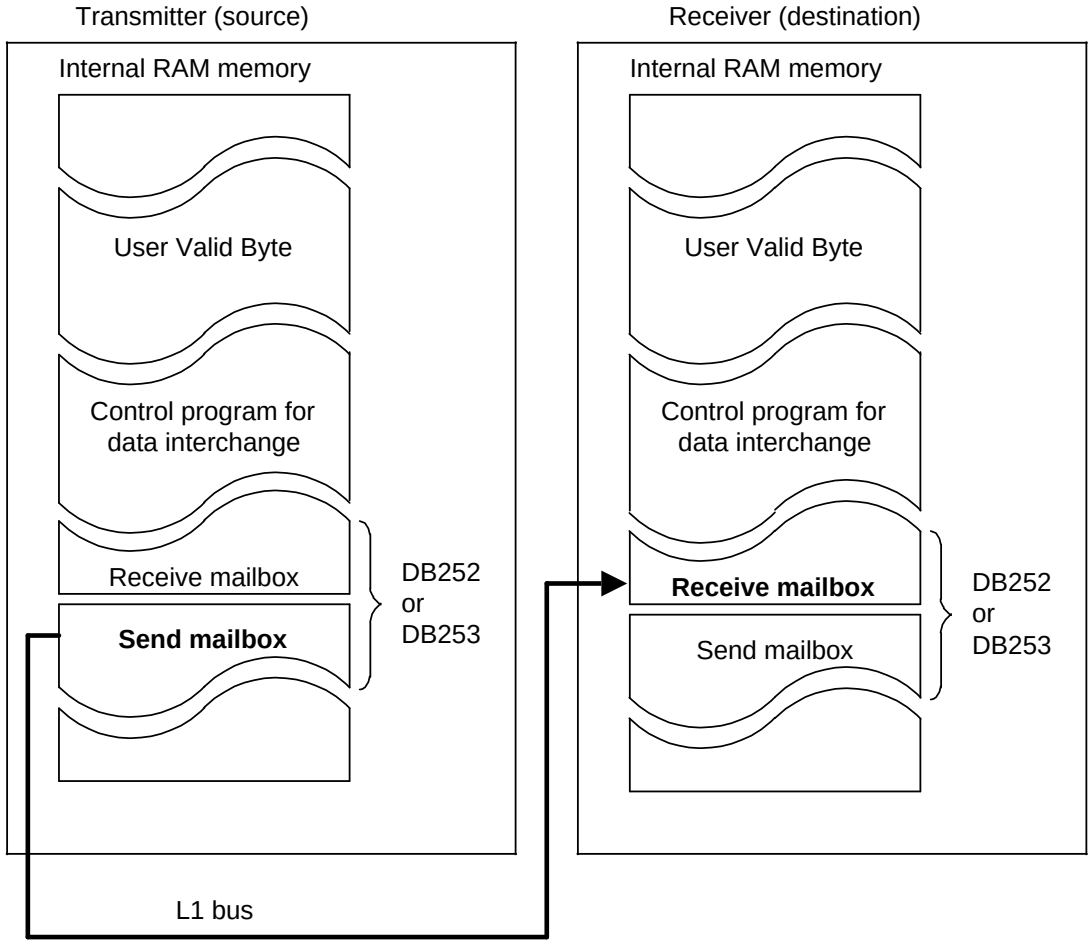
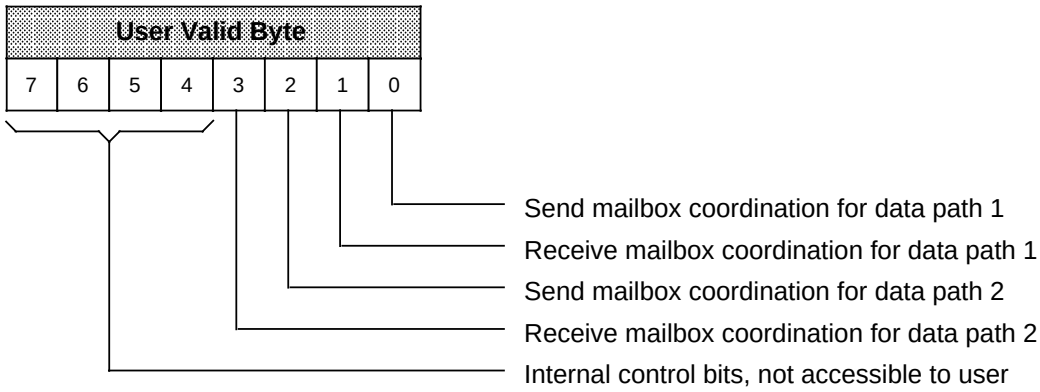


Figure 13-10. Data Interchange Between Transmitter and Receiver (Principle)

Coordinating Data Interchange via the User Valid Byte

The User Valid Byte is used to coordinate data interchange (similar to the KBE and KBS for nonfailsafe data interchange).



Coordinating the Frame Sequence (95F Mode)

The S5-95F does not transmit a "new" frame until the receiver's operating system has accepted the data in the "old" frame. The transmitter, however, receives no acknowledgement as to whether the receiver's user program evaluated the frame.

Because the S5-95F erases the contents of the Receive Mailbox in the event of a power failure, a frame is lost when the receiver's operating system received it but the user program failed to evaluate it before the power failed. If your automated process cannot tolerate frame losses, you must develop your own acknowledgement system for monitoring the frame sequence at the user level. This, in turn, requires a data path from the receiver to the transmitter.

Safety Note

SINEC L1 nodes receive no acknowledgement indicating whether a valid frame was received and subsequently evaluated by the user program. If your automated process requires failsafe frame coordination, you must use an acknowledgement system to monitor the frame sequence.

Coordination of Frame Sequence (115F-14 and 115F-15 Modes)

The S5-95F resets bit 0 or 2 of the UVB after sending the frame, even if the receiver's operating system has not accepted the data of the old frame.

Safety Note

When you use message modes 115F-14 and 115F-15, the S5-95F useful frame monitoring feature is deactivated. To prevent an undetected frame loss, you must not alter the send mailbox more than once during each safety time for Receive.

13.3.3 Sending Failsafe Data

An S5-95F can send failsafe data to no more than two slaves (two "Send" paths). The destination slaves must be either S5-95Fs or S5-115Fs. You can transfer up to 60 bytes of net data per transmission.

Each data path has a fixed mailbox. These are

- DB 252, beginning DW 32, for "Send data path 1"
- DB 253, beginning DW 32, for "Send data path 2"

The following is prerequisite to sending data:

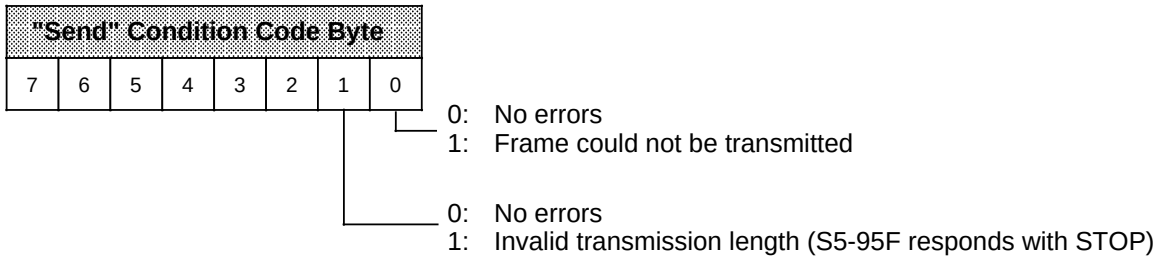
- Parameter blocks SL1: and SL2S: must be available in DB1
- You have transferred the net data and auxiliary information (number of net data bytes) to the Send Mailbox

The diagram below shows you which information is stored where in the Send Mailbox.

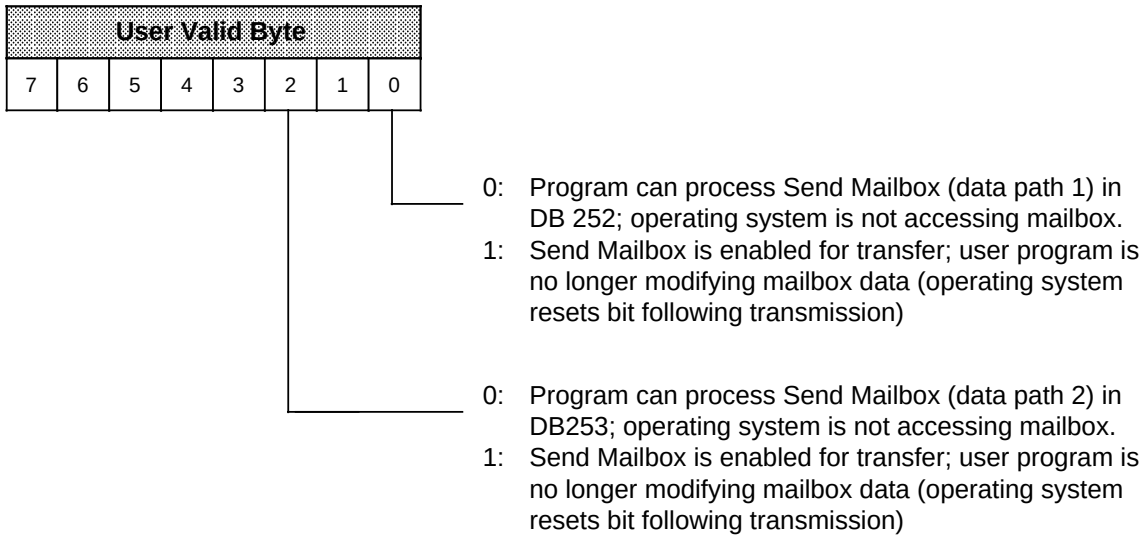
Contents of the SINEC L1 DBs (DB 252 and DB 253)

Data word	Contents of the data word	
DL	**Receive Mailbox**	**DR**
DW 0	No. of net data bytes	Receive condition code and control byte
DW 1	Net data byte 1	
DW 2	Net data byte 3	Net data byte 2
.		Net data byte 4
.		
.		
DW 30	Net data byte 59	Net data byte 60
DL	**Send Mailbox**	**DR**
DW 32	No. of net data bytes	Send condition code
DW 33	Net data byte 1	Net data byte 2
DW 34	Net data byte 3	Net data byte 4
.		
.		
.		
DW 62	Net data byte 59	Net data byte 60

Contents of the "Send" Condition Code Byte (DR 32)



Send Coordination Via the User Valid Byte (as per the Parameter Values Initialized with COM 95F)



User Program for Transmitting Data

The user program for transmitting data should have the following structure:

Scan bit 0 or bit 2 in the UVB to see if permission to send has been granted.
As long as the S5-95F is accessing the Send Mailbox, the relevant bit is set; the contents of the mailbox must not be changed during this phase.

When the operating system has reset bit 0 or bit 2 in the UVB,
- enter the number of net data bytes to be transferred into DL32
- reset the condition code byte (write 00_{H} into DR32)
- write the net data into the data words, beginning with data word DW33

Start the Send by setting bit 0 or bit 2.

When the operating has reset bit 0 or bit 2, check to see if the frame was transmitted.
The "condition code byte for Send" (DR32) contains 00_{H} if the frame was transmitted successfully (local acknowledgement).
If the frame could not be transmitted, DR32 contains 01_{H}.

13.3.4 Receiving Failsafe Data

The S5-95F can receive failsafe data from up to two slaves (two "Receive" data paths). The source PLC must be either an S5-95F or an S5-115F. Up to 60 net data bytes may be transferred each time.

Each data path has a fixed Receive Mailbox; the Receive Mailboxes are in

- DB 252, beginning data word DW 1, for data path 1
- DB 253, beginning data word DW 0, for data path 2.

Before data can be received, you must have

- initialized parameter blocks SL1: and SL1S: in DB1.

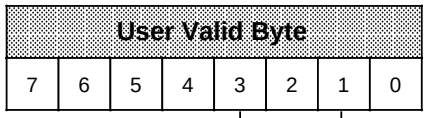
The diagram below shows which information is stored where in the Receive Mailbox.

Contents of the SINEC L1 DBs (DB252 and DB253)

Data word	Contents of the data word	
	DL Receive Mailbox	**DR**
DW 0	No. of net data bytes	Receive condition code and control byte
DW 1	Net data byte 1	Net data byte 2
DW 2	Net data byte 3	Net data byte 4
.		
.		
.		
DW 30	Net data byte 59	Net data byte 60
	DL Send Mailbox	**DR**
DW 32	No. of net data bytes	Send condition code byte
DW 33	Net data byte 1	Net data byte 2
DW 34	Net data byte 3	Net data byte 4
.		
.		
.		
DW 62	Net data byte 59	Net data byte 60

Receive Coordination Via the User Valid Byte (as per the Parameter Values Initialized with COM 95F)

You cannot process the Receive Mailbox until the operating system has completed the data transfer. To ascertain the status of the data transfer, you can evaluate the User Valid Byte.



0: Program can process Receive Mailbox (data path 1) in DB 252.
1: Program is waiting for next useful data frame; the operating system will reset this bit as soon as the next useful data frame has been transferred without error.

0: Program can process Receive Mailbox (data path 2) in DB 253.
1: Program is waiting for next useful data frame; the operating system will reset this bit as soon as the next useful data frame has been transferred without error.

User Program for Receiving Data

The user program for receiving data should have the following structure:

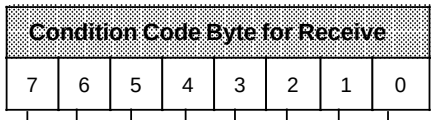
Check to see if a frame has been received.
Bit 1 or bit 3 in the UVB are "0".

Set bit 1 or bit 3 in the UVB to inform the operating system that you are ready to receive the next frame.

When the operating system resets bit 1 or bit 3, read the Receive Mailbox. Evaluate
- DL 0 (number of net data bytes received)
- DR 0 (condition code byte for Receive)
- DW 1 and subsequent data words (net data)

After you have evaluated this data, set bit 0 or bit 3 to inform the operating system that you are ready to receive the next frame.

Contents of the Condition Code and Control Byte for Receive (DR 0 in 95F Message Mode)



0: Frame valid (operating system resets bit in the RESTART routine and following depassivation via FB255)
1: Frame invalid. Safety time expired.
Depending on how DB1 was initialized, the system either reacts with a PLC STOP or responds as programmed at the user level. No other frames are transferred to the Receive Mailbox as long as this bit is "1".

0: o.k.
1: Monitoring time for data transfer via subunit A expired. Fault is tolerated on high-availability L1 bus. The operating system resets this bit when a valid frame is received.

0: o.k.
1: Monitoring time for data transfer via subunit B expired. Fault is tolerated on high-availability L1 bus. The operating system resets this bit when a valid frame is received.

0: o.k.
1: Useful frame lost; internal useful frame counter incorrectly incremented. Safety time will not be restarted as long as this bit is "1".
The operating system resets the bit following successful depassivation.

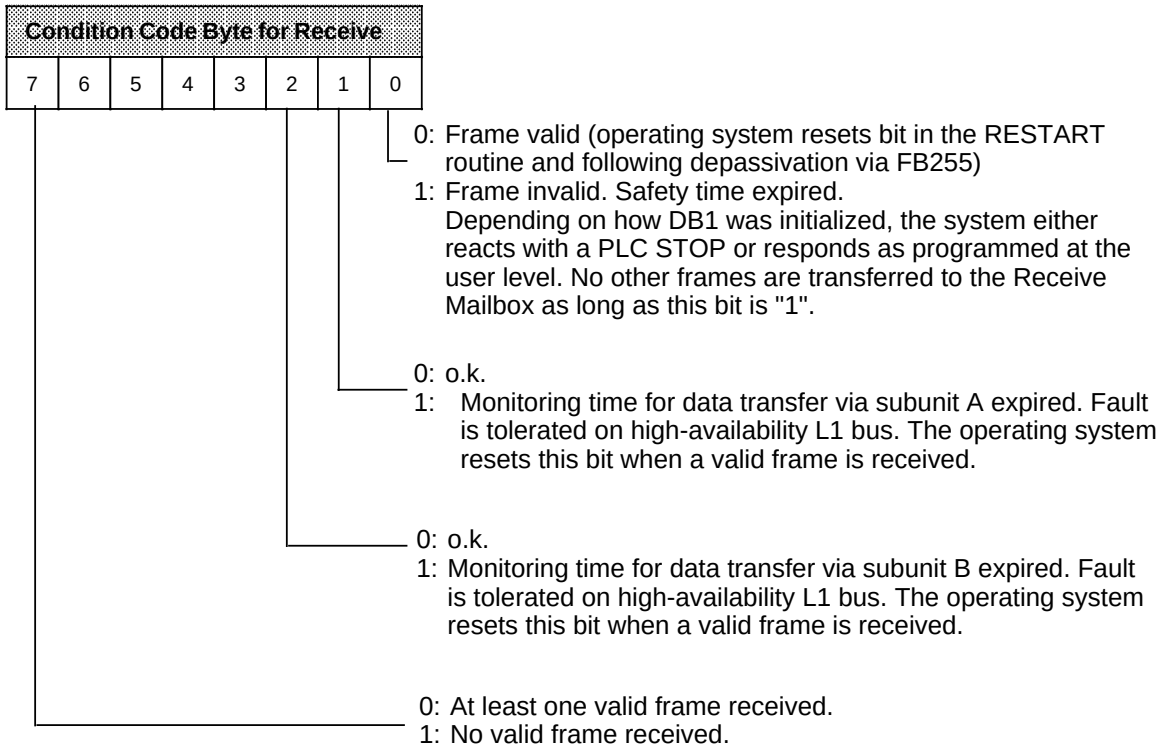
0: o.k.
1: Frame on subunit A rejected, internal change byte is not altered. The safety time is not restarted as long as this bit is "1".
The operating system resets the bit following successful depassivation.

0: o.k.
1: Frame on subunit B rejected, internal change byte is not altered. The safety time is not restarted as long as this bit is "1".
The operating system resets the bit following successful depassivation.

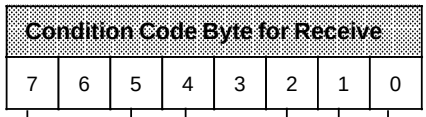
0: o.k.
1: Bit for depassivation of the data paths. When this bit is set, the internal timer which monitors the SINEC L1 safety time is restarted and bits 3, 4 and 5 reset.
Bit is reset
- by the operating system when an error-free frame was received within the specified safety time
- when the safety time has expired

0: At least one valid frame received.
1: No valid frame received.

Contents of the Condition Code and Control Byte for Receive (DR0 in 115F-14 Message Mode)



Contents of the Condition Code and Control Byte for Receive (DR0 in 115F-15 Message Mode)



0: Frame valid (operating system resets bit in the RESTART routine and following depassivation via FB255)
1: Frame invalid. Safety time expired.
Depending on how DB1 was initialized, the system either reacts with a PLC STOP or responds as programmed at the user level. No other frames are transferred to the Receive Mailbox as long as this bit is "1".

0: o.k.
1: Monitoring time for data transfer via subunit A expired. Fault is tolerated on high-availability L1 bus. The operating system resets this bit when a valid frame is received.

0: o.k.
1: Monitoring time for data transfer via subunit B expired. Fault is tolerated on high-availability L1 bus. The operating system resets this bit when a valid frame is received.

0: o.k.
1: Frame on subunit A rejected, internal change byte is not altered. The safety time is not restarted as long as this bit is "1".
The operating system resets the bit following successful depassivation.

0: o.k.
1: Frame on subunit B rejected, internal change byte is not altered. The safety time is not restarted as long as this bit is "1".
The operating system resets the bit following successful depassivation.

0: At least one valid frame received.
1: No valid frame received.

13.4 SINEC L1 Safety Times

On a failsafe SINEC L1, the S5-95F differentiates between two types of safety time: The SINEC L1 safety time for Receive and the SINEC L1 safety time for Send. The safety times may be specified separately for each data path (i.e. as many as four different safety times may be selected).

13.4.1 SINEC L1 Safety Time for Receive

Definition: The SINEC L1 safety time for Receive defines a time period for failsafe data paths during which the interfaced node must receive at least one valid frame.

The SINEC L1 safety time for Receive is dependent on the automated process and on the peers' control program. It requires approval, and must be endorsed by the inspector at the time of the selective acceptance test.

At that time, you must agree with the inspector on

- the SINEC L1 safety time for Receive and
- how the system is to respond when no valid frame is received within that time period

for each failsafe data path.

> **Note**
>
> It is not necessary to transmit and receive at least one useful frame within the SINEC L1 safety time. If need be, the operating system transmits additional test frames, unbeknownst to the user, for monitoring the bus.

Interval for Receiving Frames

> **Note**
>
> Please note that the S5-95F cannot receive a new frame until it has processed the previous one. You should therefore arrange communications between nodes so that frames are received at least 100 ms apart, otherwise the S5-95F may not accept the next frame.

Conditions for the SINEC L1 Safety Time for Receive

1st condition

The first condition is based upon the principles of the transmitting and receiving system.

Fulfillment of this condition is the basis of the SINEC L1 safety time for the receiving S5-95F:

$$\text{SINEC L1 safety time for Receive} \quad \frac{2\cdot(k+l)\cdot\text{SINEC L1 bus cycle time}}{n} + 200\ \text{ms}$$

k= Number of failsafe Send data paths over which the S5-95F (k=1 or 2) or S5-115F (k=1 to 30) transmitter sends data;
- k=1 for one failsafe data path
- k=2 to 30 for 2 to 30 failsafe data paths (the data path to the master must also be included in the case of the S5-115F)

l= Factor for nonfailsafe data paths;
- l=0, when the transmitter never or very rarely sends nonfailsafe data
- l=1, when the transmitter sends nonfailsafe data frequently

n= Number of times the transmitter appears in the SINEC L1 polling list
If the transmitter appears more than once, it must be evenly distributed throughout. A sample polling list in this case might be: 1 **2** 3 4 **2** 5 6 **2** 7 8 **2** 9

SINEC L1 bus cycle time = Frame size (plus 4 bytes in the case of failsafe frames)×2 ms
\+ Number of failsafe data paths in the whole SINEC L1 network×44 ms

TIP: You can reduce the bus cycle time by reducing the amount of data and/or data paths in the polling list.

2nd condition

The minimum time period between two successive frames from the master to its slave must be 1.5 · the slave's SINEC L1 safety time for Receive.

3rd condition (for double SINEC L1 bus only)

When the S5-95F receives frames from the master, the following condition must be met when using a double SINEC L1 channel:

SINEC L1 safety time for Receive SINEC L1 safety time for Send + 500 ms

For information on computing the SINEC L1 safety time for Send, see section 13.4.2.

Error Response from Slave Node

The following steps are carried out if a node does not receive a valid frame within the stipulated SINEC L1 safety time for Receive:

- The operating system sets bit 0 in the "condition code and control byte Receive" and clears the Receive mailbox
- The operating system enters the error in the system event DB254's SINEC L1 error word
- The operating system sets an error flag in the system event DB254's error buffer
- The S5-95F branches to the error response programmed in DB1 (possible reactions to an error are either a PLC STOP or a reaction invoked at the user level in the user program).

Problems with the SINEC L1 Bus Master

When problems occur with the SINEC L1 bus master, e.g. because it has gone to STOP or because it is restarting following a POWER OFF-POWER ON sequence, they may cause a frame delay. Should this happen, the SINEC L1 safety time could expire. In such a situation, the S5-95F responds with STOP or with passivation of the data path, depending on the parameter values specified in DB1.

In 95F message mode you can depassivate the passivated data path by setting bit 6 of the ”condition code and control byte for Receive” after taking steps to ensure safety.

In 115F-14 or 115F-15 message mode the operating system automatically depassivates the data path on receipt of a valid frame.

13.4.2 SINEC L1 Safety Time for Send

The S5-95F controls transmission of failsafe frames via an internal timer. The S5-95F always restarts the timer as soon as it expires. As long as the internal timer is running, the S5-95F transmits (during one timer cycle) the failsafe frames over data path 1 and data path 2; then, if enough time is left, it transmits nonfailsafe frames.

The S5-95F computes the initial value of the internal timer from the smaller of the two SINEC L1 safety times for Send (data paths 1 and 2) as follows:

Internal timer = 0.5 x SINEC L1 safety time for Send - 100 ms

Conditions for the SINEC L1 Safety Time for Send

The SINEC L1 safety time for Send must fulfill the following conditions:

1st condition

$$\text{SINEC L1 safety time for Send} \quad \frac{2\cdot(k+l)\cdot\text{SINEC L1 bus cycle time}}{n}$$

k= Number of failsafe data paths initialized for Send, where
- k=1 for one failsafe data path
- k=2 for two failsafe data paths

l= Factor for nonfailsafe data traffic, whereby
- l=0, when nonfailsafe data is never or only rarely transmitted
- l=1, when nonfailsafe data is transmitted frequently

n= Number of occurrences of the S5-95F (station number of the transmitter) in the SINEC L1 polling list
If the S5-95F appears more than once in the polling list, its station number must be evenly distributed throughout the list. A sample polling list might be: 1 **2** 3 4 **2** 5 6 **2** 7 8 **2** 9

SINEC L1 bus cycle time = Number of frame bytes (plus 4 bytes in the case of failsafe frames)×2 ms
\+ Number of failsafe data paths in the entire SINEC L1 network×44 ms

2nd condition

The SINEC L1 safety time for Send must be 300 ms.

3rd condition (for double SINEC L1 bus only)

The following condition must be fulfilled when using a double SINEC L1 channel:

SINEC L1 safety time for Receive SINEC L1 safety time for Send + 500 ms

4th condition (for single SINEC L1 channel only)

The following condition must be fulfilled when using a single SINEC L1 channel:

SINEC L1 safety time for Receive SINEC L1 safety time for Send +200 ms

13.4.3 Load Placed on the System by the SINEC L1 Channel

PLC Onload Due to Traffic on the SINEC L1 Channel

Traffic on the SINEC L1 channel increases the load on the PLC cycle. The load caused by SINEC L1 traffic

- Increases in proportion to the number of frames and the frame lengths
- Is twice as high when a double bus is used

The load is due to frame backup (0.02 ms per byte), in addition to the base load (0.06 ms per byte). It amounts to approx. 0.08 ms per SINEC L1 channel for each failsafe data byte transmitted and approx. 0.06 ms for each non-failsafe data byte.

The relative system onload caused by traffic on the SINEC L1 channel is

- Max. 8 % for a single SINEC L1 channel
- Max. 12 % for a double SINEC L1 channel (see section 7.4.2)

Minimizing the Relative System Onload

The statements made above apply when the user program utilizes to the full the transfer options provided by the SINEC L1 bus. In most cases, this is not necessary.

You can reduce the relative system onload caused by traffic over the SINEC L1 channel by doing the following:

- Select the longest time period permitted by the process as SINEC L1 safety time for Send.

- When the user program is transmitting ***and*** last sent a long frame, send a short blank frame, but wait one SEND safety-time period.
  Reason: In order to prevent frame loss, the operating system sends another frame with the same contents within the safety time period as long as there is no further frame that needs to be transmitted. The test frames which then follow the blank frame "switch" bus traffic to minimum load until the next long user frame is ready to be sent.

TIP: Keep frames as short as possible. If you can, you should edit and optimize the data to be transmitted (data compression), even when this increases program runtimes.

13.4.4 Response Time During SINEC L1 Traffic

The response time during SINEC L1 traffic is the time from the input signal change in the transmitting system to the output signal change in the receiving system.

The response time is a combination of three time periods:

- Processing time in the transmitter (time between an input signal change and updating of the SINEC DB, including the Send request in the User Valid Byte)
- SINEC L1 safety time for Receive
- Processing time in the receiver (time between evaluation of the SINEC L1 DB, including the Receive request in the User Valid Byte, and the output signal change).

13.4.5 Transmission of Error Messages to the SINEC L1 Master

The S5-95F can automatically transmit system messages to the SINEC L1 master of channel B. This enables you to evaluate, display and print out all messages from a central point.

If the appropriate parameters are entered in DB1 (see COM 95F manual), the S5-95F will send every message entered in the error stack of DB254 to the SINEC L1 master of channel B. Nonfailsafe frames to the SINEC L1 master have a fixed length of 16 net data bytes. The SINEC L1 master can identify frames as system messages from the source slave number and the frame length.
Avoid sending user frames with a length of 16 net data bytes to the SINEC L1 master because it will interpret them as system messages.

13.4.6 Example: SINEC L1 Cycle Time and SINEC L1 Safety Time

The following section demonstrates the relationship between the SINEC L1 cycle time and the SINEC L1 safety times for Send and Receive.

Description:

In the example a master and three slaves are connected to the SINEC L1 LAN. Slave 1 and Slave 3 are S5-95F devices and slave 2 is an S5-115F programmable controller with CPU 942-7UF15.

Data traffic over the double SINEC L1 bus is arranged so that slave 1 can both send and receive via data paths 1 and 2 to slaves 2 and 3. Nonfailsafe data traffic between the two masters and all three slaves is also required.

Figure 13-11 shows the structure of the SINEC L1 network.

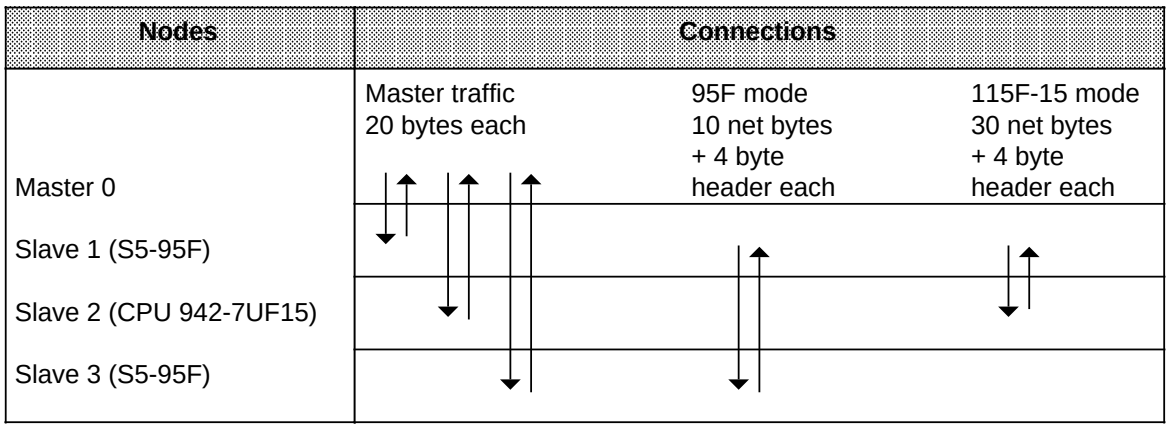


Figure 13-11. Network Structure

Specifying the Polling List

Master traffic must be guaranteed in each bus cycle. This results in the following sequence for the polling list in the SINEC L1 bus master:

Polling list: 1-2-3-1-2-1-3

Calculating the SINEC L1 Bus Cycle Time

The SINEC L1 bus cycle time is calculated as described in section 13.4.1:

Quantity of data transmitted = 6 x 20 bytes+2 x (10+4) bytes+2 x (30+4) bytes+= 216 bytes

SINEC L1 bus cycle time = 216 x 2 ms + 44 x 7 ms= 740 ms

Calculating the SINEC L1 Safety Time

The SINEC L1 safety time depends on the process being automated and is agreed between you and the inspector at the selective acceptance test. You must determine the SINEC L1 safety time actually required for data traffic by performing the following calculation. The safety time required must be shorter than the SINEC L1 safety time specified by the inspector.

The SINEC L1 safety time actually required for the specified data transfers is determined as follows:

SINEC L1 Safety Time for Send for Slave 1:

The SINEC L1 safety time for Send must conform with the defined conditions (see section 13.4.2).

1st condition

$$\text{SINEC L1 safety time for Send} \quad \frac{2 \cdot (k+l) \cdot \text{SINEC L1 bus cycle time}}{n}$$

where k = 2 data paths
l = 1 (nonfailsafe data transmitted frequently)
n = 3 occurrences in the polling list
yields the result:

$$\text{SINEC L1 safety time for Send} = \frac{2 \ (2 + 1) \times 740}{3} \text{ ms} = 1480 \text{ ms}$$

2nd condition

The SINEC L1 safety time for Send must be 300 ms. This condition is fulfilled.

3rd condition (for double SINEC L1 channel only)

The permissible SINEC L1 safety time for Receive must be agreed with the inspector. The following condition must be fulfilled:

SINEC L1 safety time for Receive SINEC L1 safety time for Send +500 ms.

SINEC L1 Safety Time for Send for Slave 2 and Slave 3:

The SINEC L1 safety time for Send must conform with the defined conditions (see section 13.4.2).

1st condition

$$\text{SINEC L1 safety time for Send} \quad \frac{2 \cdot (k+l) \cdot \text{SINEC L1 bus cycle time}}{n}$$

where k = 1 data paths
l = 1 (nonfailsafe data transmitted frequently)
n = 2 occurrences in the polling list
yields the result:

$$\text{SINEC L1 safety time for Send} = \frac{2\ (1 + 1) \times 740}{2} \text{ ms} = 1480 \text{ ms}$$

2nd condition

The SINEC L1 safety time for Send must be 300 ms. This condition is fulfilled.

3rd condition (for double SINEC L1 channel only)

The permissible SINEC L1 safety time for Receive must be agreed with the inspector. The following condition must be fulfilled:

SINEC L1 safety time for Receive SINEC L1 safety time for Send +500 ms.

13.5 Connecting the S5-95F to the PROFIBUS

The S5-95F can also be connected to the PROFIBUS with the help of a CP 541 communications processor. To do so, you must connect the CP 541 to the serial port of one of the subunits.

For the S5-95F, the CP 541 has the same performance characteristics as a complete SINEC L1 bus.

The CP 541 supports the simultaneous use of different PROFIBUS communications modes:

- Failsafe PLC-PLC connection to S5-95F and S5-115F
- Failsafe Multicast to S5-95F and S5-115F via FDL
- Non-failsafe connections to DP nodes
- Non-failsafe PLC-PLC connection

You will find additional information on the use of the CP 541 communications processor in the CP 541 manual.

Example for Connecting the S5-95F to the PROFIBUS

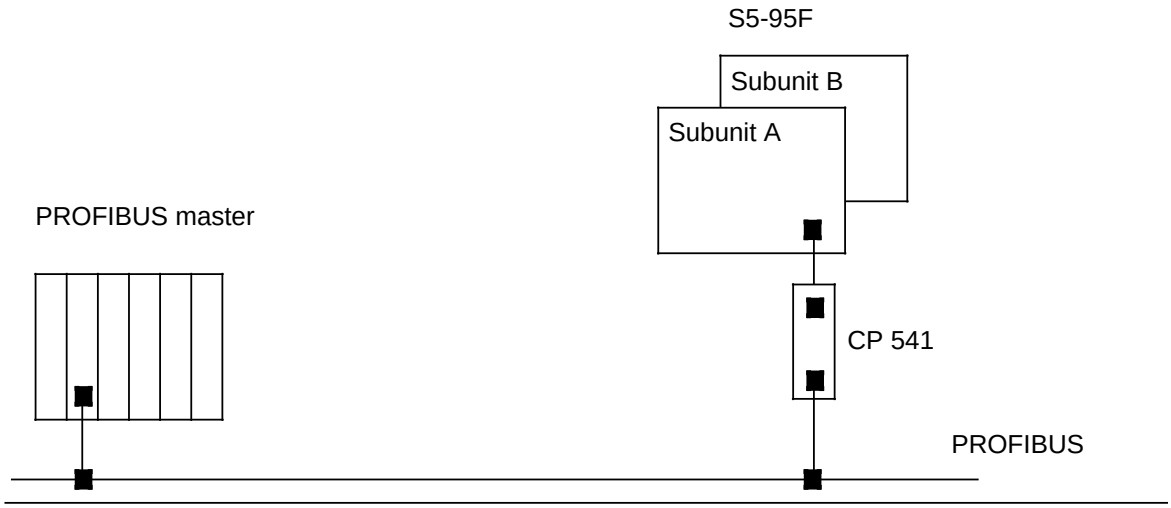


Figure 13-12. Subunit A via CP 541 to the PROFIBUS

Note

Note that the CP 541 does not support the SINEC L1's PG bus function.

14 Testing the User Program and Storing It on the Memory Submodule

14.1	Testing and Debugging the User Program	14- 1
14.1.1	Program-Dependent Signal Status Display "STATUS"	14- 1
14.1.2	Direct Signal Status Display "STATUS VAR"	14- 3
14.1.3	Forcing Variables with "FORCE VAR"	14- 4
14.1.4	Search Function	14- 4
14.2	Interrupt Analysis with the Programmer	14- 5
14.2.1	The "ISTACK" Analysis Function	14- 5
14.2.2	Descriptions of the ISTACK Flags	14- 9
14.3	Program Errors	14- 10
14.3.1	Determining the Error Address	14- 10
14.3.2	Program Trace with the "BSTACK" Function	14- 12
14.4	Programmer Control Functions	14- 13
14.5	Measures for Securing the User Program Against Errors	14- 15
14.6	Storing the User Program on Memory Submodules	14- 16

Figures

14-1 "STATUS" Test Function 14- 1
14-2 "STATUS VAR" Test Function 14- 3
14-3 Contents of the Interrupt Condition Code Word 14- 7
14-4 Reloading and Compiling a Block in RUN Mode 14- 11
14-5 Program Trace with the "BSTACK" 14- 12

Tables

14-1 Restrictions on the "STATUS" Function 14- 2
14-2 Control Bits 14- 6
14-3 Interrupt Stack 14- 6
14-4 ISTACK Entries in System Data Words 203 to 214 . . . . . 14- 7
14-5 Abbreviations for Control Bits and Causes of Faults 14- 8
14-6 Interrupt Analysis 14- 9
14-7 Operating the Programmer in Safety, Quasi-Safety and Test Mode 14- 14
14-8 Storing a User Program on EPROM 14- 16
14-9 Blocks Which Must be Stored on the EPROM 14- 17

14 Testing the User Program and Storing It on the Memory Submodule

This section provides:

- A summary of the most important functions for debugging the user program
- Information on how to transfer and save the debugged user program on the EPROM submodule

14.1 Testing and Debugging the User Program

The STEP 5 basic package provides functions for debugging and testing your control program. In the following sections, you will find a brief summary of the most important functions for locating errors in the program logic. A detailed description of these test functions is included in the manual which accompanies your programmer.

14.1.1 Program-Dependent Signal Status Display "STATUS"

This test function displays the current signal states and the RLO (result of the logic operation) for the operations executed during the program scan.
You can also use this and other test functions to debug the program.

> **Note**
>
> The current signal states are displayed in "RUN" mode only. Note that the programmer's STATUS function increases the relevant block's runtime, and thus the scan time. If a sufficiently long scan time was not set, the S5-95F could, under certain circumstances, enter the STOP mode.

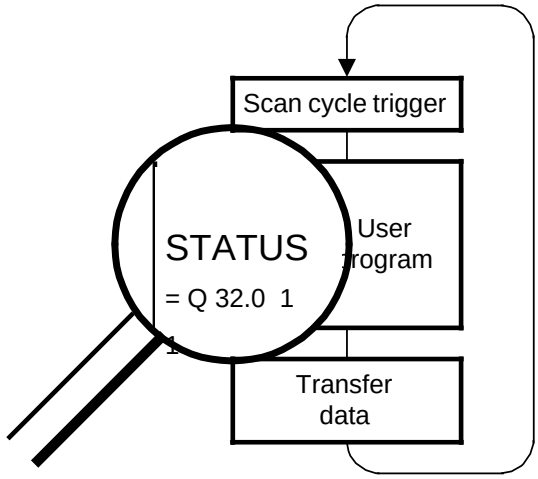


Figure 14-1. "STATUS" Test Function

Special Features of the STATUS Function

The STATUS function increases the program scan time. The amount of the increase depends on the statements used. To prevent the program scan time from being exceeded because of STATUS, the specified scan monitoring time should be as long as the process response time allows.

Restrictions on the STATUS Function

The STATUS function is restricted for individual program sequences on the S5-95F.

Table 14-1. Restrictions on the "STATUS" Function

STATUS for ...	Test Mode	Quasi-/Safety Mode
OB1 and OB13	No restrictions	No STATUS for blocks invoked between IA and RA operations
OB2	STATUS not possible	STATUS not possible
OB3	No restrictions	STATUS not possible

14.1.2 Direct Signal Status Display "STATUS VAR"

This test function returns the status of an arbitrary operand (input, output, flag, data word, counter or timer) at the end of the program scan. Information about inputs and outputs can be obtained from the process input and process output images.

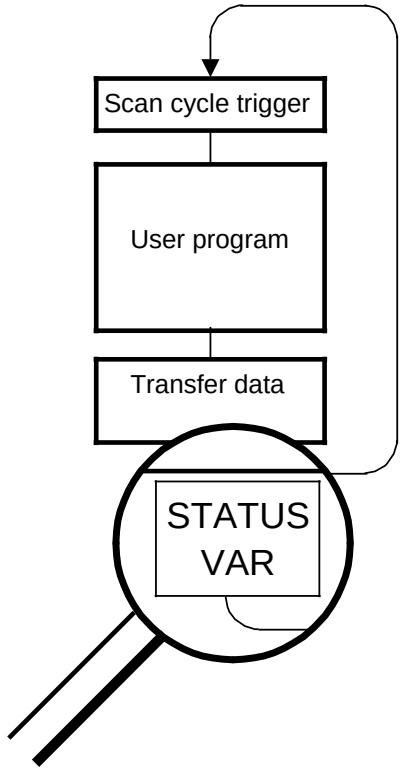


Figure 14-2. "STATUS VAR" Test Function

14.1.3 Forcing Variables with "FORCE VAR"

FORCE VAR allows you to modify the following variables when the programmable controller is either at STOP or RUN: I, Q, F, T, C and D.

When the PLC is at RUN, the new process variables are used for the program scan. The variables can be changed again, without any indication, in the course of the program. The process variables are controlled asynchronously to the program scan, and in both subunits simultaneously.

Special Features:

- Changes are possible in debug mode only; in failsafe or quasi-failsafe mode, they may only be read.
- I, Q and F variables may be changed in the process I/O image either bit by bit, byte by byte or word by word.
- Edge flag control must be observed in the case of T and C variables in KM format.
- The signal status display is aborted if a bad format or operand entry is detected, and the programmer outputs the "NO FORCING POSSIBLE" message.

Information on invoking the test functions on the programmer can be found in the relevant programmer manuals.

14.1.4 Search Function

Search locates specific "keys" in the program and lists them on the programmer's display. Once found, the "keys" may be modified.

A search may be instigated in the following programmer functions:

- INPUT
- OUTPUT
- STATUS

Possible search keys are:

- Statements (e.g. A I 32.0)
- Operands (e.g. Q 32.5)
- Labels (e.g. X01) Possible in function blocks only!
- Addresses (e.g. 0006_H)

> **Note**
>
> Different programmers execute a search in different ways. For details, please refer to the programmer operating manual.

14.2 Interrupt Analysis with the Programmer

When malfunctions occur, the operating system sets various "analysis bits"; you can then scan these bits using the programmer's "ISTACK" function.

14.2.1 The "ISTACK" Analysis Function

The interrupt stack is located in internal memory. The S5-95F uses the ISTACK to flag the cause of a fault. When a fault occurs, a bit is set in the relevant byte of the ISTACK.

The interrupt stack can be read out via the programmer.

Invoking the ISTACK

The ISTACK is invoked via the programmer menu while the PLC is at "STOP".
Refer to the programmer manual for the keystroke sequence.

ISTACK Updating

The S5-95F updates the ISTACK

- when the PLC goes from "RUN" to "STOP" without actuation of the mode switch (RUN STOP)
- when the S5-95F is switched from "STOP" to "RUN" but still remains at "STOP"

Redisplay the contents of the ISTACK on the programmer to ascertain the current cause of error.

The tables on the following pages show:

- The control bits set in the ISTACK
- The causes of error flagged in the ISTACK
- The system data words in which the ISTACK flags are stored and
- Descriptions of the abbreviations and error flags used.

ISTACK Output on the PG 710/730/750 and 770 Programmers

Table 14-2 shows a programmer listing of ISTACK control bits. The control bits needed for fault analysis are emphasized in bold type.

The righthand portion of the table shows the allocation of control bits to system data and absolute addresses in the PLCs.

Table 14-2. Control Bits

CONTROL BITS							
NB	PBSSCH	BSTSCH	SCHTAE	ADRBAU	SPABBR	NAUAS	QUITT
NB	NB	NB	REMAN	NB	NB	NB	NB
STOZUS	STOANZ	NEUSTA	NB	BATPUF X	NB	BARB	BARBEND
NB	UAFEHL	MAFEHL	EOVH	NB	AF	NB	NB
ASPNEP	ASPNRA	**KOPFNI**	PROEND	ASPNEEP	PADRFE	ASPLUE	RAMADFE
KEINAS	**SYNFEH**	**NINEU**	NB	NB	NB	SUMF	URLAD

System data word (SD)	Absolute S5-95F address
SD 5	$5D0A_H$
SD 6	$5D0C_H$
SD 7	$5D0E_H$

Table 14-3 shows the programmer ISTACK listing for the causes of errors. Causes of error required for S5-95F fault analysis are emphasized in bold type.

Table 14-3. Interrupt Stack

INTERRUPT STACK							
DEPTH:	01						
BEF-REG:	0000	SAC:	25CA	DB-ADR:	0000		
BST-STP:	5E03	OB-NO.:	1	DB-NO.:			
		REL-SAC:	0000				
ACCU1:	FFFF	ACCU2:	0000				
CONDITION CODE BITS:		CC1	CC0	OVFL	CARRY	OR	$\overline{\text{ERAB}}$ X
		STATUS	RLO X				
CAUSE OF FAULT:		**STOPS** X	**NB**	**SUF**	**TRAF**	**NNN**	**STS**
		STUEB	**NAU**	QVZ	**ZYK**	**PEU**	**BAU**
		ASPFA					

The ISTACK comprises system data words SD 203 to SD214. Table 14-4 shows which system data word contains which ISTACK entry.

Table 14-4. ISTACK Entries in System Data Words 203 to 214

System Data Word (SD)	ISTACK Entry	Absolute Address S5-95F
SD 203	ACCU1	$5E96_H$
SD 204	ACCU2	$5E98_H$
SD 205	Instruction register	$5E9A_H$
SD 206	Step address counter (SAC)*	$5E9C_H$
SD 207	Block stack pointer	$5E9E_H$
SD 208	Data block start address	$5EA0_H$
SD 209	Nesting depth (0 to 6) and 1st nesting level	$5EA2_H$
SD 210	2nd and 3rd nesting level	$5EA4_H$
SD 211	4th and 5th nesting level	$5EA6_H$
SD 212	6th nesting level	$5EA8_H$
SD 213	Condition code bits	$5EAA_H$
SD 214	Interrupt condition code word (see Figure 14-3)	$5EAC_H$

***** Contains the absolute memory address of the next statement to be executed or the block start address of the errored block. A parameter assignment error is indicated if the SAC points to an address in DB1 (see section 9.1). DB1 is specified as block in the ISTACK.

Contents of the Interrupt Condition Code Word (UAW)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
STOPS	u	SUF	TRAF	NNN	STS	STUEB	u	NAU	u	u	ZYK	u	PEU	BAU	ASPFA	**SD 214**

u: Unassigned

Figure 14-3. Contents of the Interrupt Condition Code Word

Table 14-5 provides a list of abbreviations used in the programmer's ISTACK listing.

Table 14-5. Abbreviations for Control Bits and Causes of Faults

Abbreviations for Relevant Control Bits		Abbreviations for Relevant Causes of Faults	
BSTSCH	Block Shift requested	NAU	Power failure
SCHTAE	Block Shift in progress (COMP:PC function)	ZYK	Cycle time exceeded. The maximum permissible program scan time was exceeded.
ADRBAU	Address list structure	PEU	I/O not ready. Power failure in I/O expansion unit; connection to I/O expansion unit interrupted; no connector in basic unit
NAUAS	Mains power failure		
STOZUS	STOP state (external request, e.g. via PG)		
STOANZ	STOP flag	BAU	Battery failure
NEUSTA	PLC cold restart in progress	ASPFA	Illegal memory submodule
BATPUF	Backup battery is OK		
AF	Interrupt Enable	Miscellaneous abbreviations:	
KOPFNI	Block header cannot be interpreted	BEF-REG:	Instruction register
		SAC	Step address counter
ASPNEEP	Memory submodule is EEPROM	DB-ADR	Data block address
KEINAS	No or too little user memory	BST-STP	Block stack pointer (points to last entry block stack)
SYNFEH	Synchronization error (block errors, e.g. bad block header)	NR	Block number (OB, PB, FB, SB, DB)
NINEU	Cold restart not possible	REL-SAC	Relative step address counter
URLAD	Bootstrapping required	CC1	See Chapter 8, "Condition Code Generation"
STOPS	Mode switch set to STOP		
SUF	Substitution error	CC0	See Chapter 8, "Condition Code Generation"
TRAF	Transfer error on data block instruction. Data word number > data block length	OVFL	Arithmetic overflow (+ or -)
NNN	PLC cannot interpret instruction	ODER	OR memory (AND before OR)
STS	Operation interrupted by PG Stop request or programmed Stop statements	ERAB	First input bit scan by pulse
		STATUS	Status of the operand of the last binary operation executed
STUEB	Block stack overflow. Maximum block call nesting depth (16) exceeded	RLO	Result of the logic operation
		u	Unassigned

14.2.2 Descriptions of the ISTACK Flags

Table 14-6 shows the causes of error which can cause interruption of a program scan. The PLC goes to "STOP" in all cases.

Table 14-6. Interrupt Analysis

ISTACK Flags	Cause of Error	Corrective Measures
ASPFA and **KEINAS** **NNN** SAC=FFFF	Program transfer error Overflow of the internal program memory during compilation. PG PLC or memory submodule PLC	Shorten program, compress memory
BAU	During automatic program loading • No battery/no charge and no valid program on memory submodule	Replace battery and rewrite or reload program
NAU	Power failure	
NINEU	Program in PLC memory is defect. Cause: • Power failure occurred during: - Memory compression - A block transfer from PG to PLC or from the memory submodule to the PLC - An overall PLC reset • Battery was changed during a power outage	Execute overall reset and reload program
NNN	• Non-decodable instruction • Nesting level exceeded • Parameter assignment error	Eliminate program error
PEU	• Expansion module not connected • I/O bus fault • Maximum length of shift register exceeded • Unknown module • Module in wrong slot	• Check power supply in expansion unit • Check all connections • Check module slots • Replace defective modules/bus modules
STOPS	Mode switch set to STOP	Set PLC to RUN
STS	• Software STOP programmed by user (STP or STS) • Stop request issued via programmer	
STUEB	Block stack overflow. Maximum block nesting depth error (16) exceeded.	Eliminate program error

Table 14-6. Interrupt Analysis (continued)

ISTACK Flags	Cause of Error	Corrective Measures
SUF	Substitution error: • Function block call with bad actual parameters • Interrupt-driven and time-controlled scanning: Integral FB call while other integral FB is being processed	Change actual parameters Disable interrupts
TRAF	Transfer error: • Data block operation programmed in which data word number exceeds data block length • Data block operation programmed without first opening DB	Eliminate program error (refer to the Programmer Manual)
ZYK	Cycle time exceeded: The program scan time exceeds the cycle monitoring time. Causes: • Program too long • Too many interrupts	Check program for continuous loops and/or shorten program

14.3 Program Errors

Two types of program errors are flagged in the ISTACK:

- Errors detected by the compiler during program compilation (compiler error "NNN")
- Errors detected during the program run (runtime errors "SUF" and "TRAF")

14.3.1 Determining the Error Address

Program Errors Detected During Compilation

When a compiler error occurs, the error address provided in the ISTACK is correct, i.e. the STEP address counter points to the absolute memory address of the STEP 5 statement immediately preceding the statement which caused the S5-95F to go to "STOP".

The relative STEP address counter (REL-SAC) gives the address of the STEP 5 statement immediately preceding the statement which caused the PLC to go to STOP relative to the start of the block.

Example: Reloading and Compiling a Block in RUN Mode

An illegal statement was programmed in PB7 and was detected by the compiler.

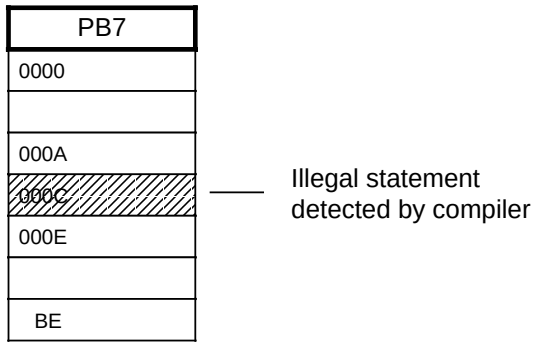


Figure 14-4. Reloading and Compiling a Block in RUN Mode

When it encounters an illegal statement, the S5-95F interrupts the program scan and goes to "STOP" with error "NNN".

The STEP address counter is set to the ***absolute address*** of next statement in program memory.

The REL SAC is set to the ***relative address*** of the next statement in PB7 ($000E_H$).

Program Errors Occurring During Program Scanning (Runtime Errors)

If errors occur during program scanning without the programmer's STATUS function having been used, the STEP address counter is set to the ***start address of the errored block***.

The ***relative address counter*** (REL-SAC) is set to "0000_H". A more precise error location is not possible.

For runtime errors such as "SUF" and "TRAF", however, you can ascertain the exact error address in a somewhat roundabout way.

To do so, proceed as follows:

Set the S5-95F to "STOP".
On the programmer, invoke the "STATUS" function for the errored block.
Switch the S5-95F from "STOP" to "RUN".
When the errored block is rescanned, the S5-95F once again goes to STOP; you can now evaluate the ISTACK on the programmer.

The STEP address counter now shows the absolute memory address, and the ***relative STEP address counter*** (REL-SAC) shows the relative address within the block preceding the address at which the S5-95F went to "STOP".

14.3.2 Program Trace with the "BSTACK" Function

During program scanning, the following information is entered in the block stack (BSTACK) for each block called:

- Block type and block number
- Absolute block start address. The absolute block start address is the start address of the block in the program memory.
- Absolute return address. The absolute return address is the memory address at which the block was exited.
- Relative return address. The relative return address is the relative address at which the block was exited.
- Data block valid when the block was exited.

This information can be called up via the programmer's "BSTACK" function while the PLC is at "STOP", but only if the PLC went to "STOP" because of a fault. "BSTACK" then shows the contents of the block stack at the time of interruption.

Example: The program scan was interrupted in FB2, and the PLC went to STOP with a "TRAF" error (illegal DB access operation, e.g. DB5 is two words long, DB3 ten words long).

The "BSTACK" function helps the user determine the path the program took to reach FB2, and tells him which DB was open at the time FB2 was invoked. It contains the three (marked) block start addresses.

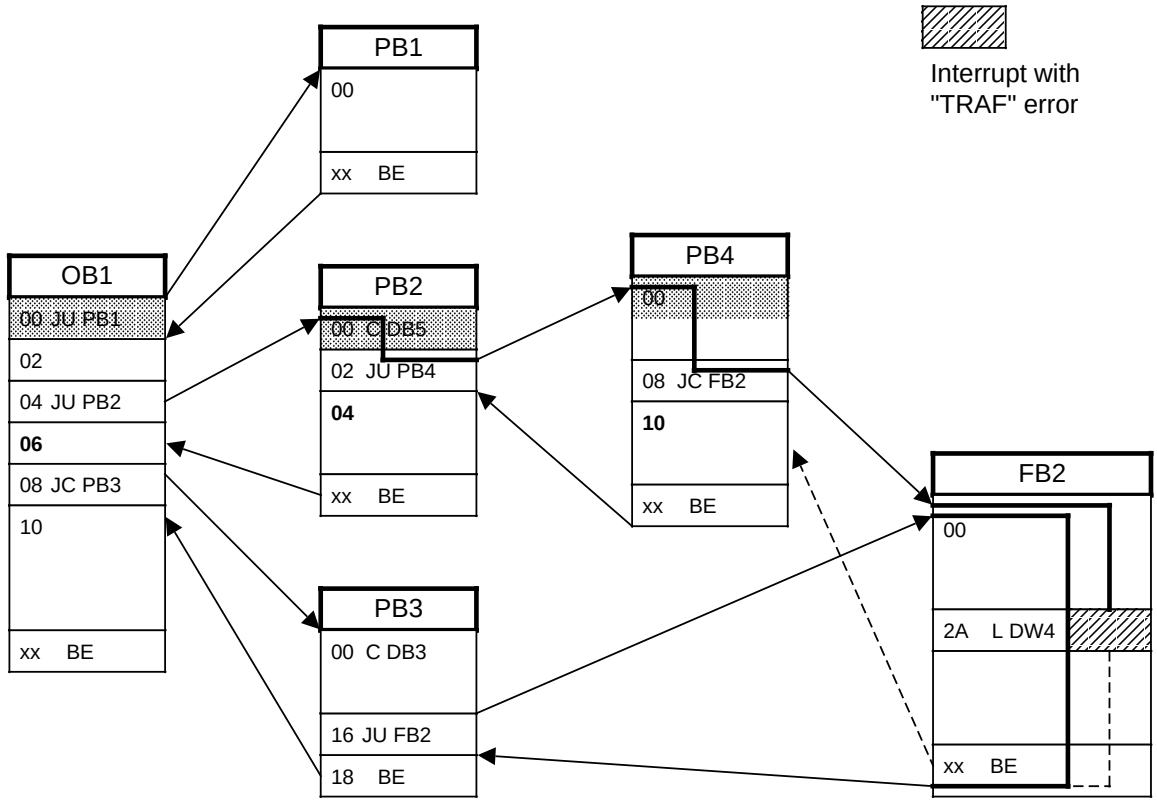


Figure 14-5. Program Trace with the "BSTACK"

14.4 Programmer Control Functions

For safety reasons, the programmer control functions are restricted in failsafe systems. In such systems, only read-only programmer functions may be invoked.
The only exception is FB235's input DB and the parameter control DB, which can be both read and modified even in failsafe mode via programmers/TDs/OPs.

FB235's Input DB

In failsafe RUN mode, you can modify the input DB specified in FB235 via a text display or operator panel.

Because the data transfer from TD/OP to the S5-95F is always non-interacting, the input data must be treated as non-safe data. Before processing the input data in the user program, you must first subject the data to plausibility and safety range checks in a so-called filter routine.

The filter routine must be approved just like any other user program block by the person conducting the acceptance test.

Parameter Control DB

In failsafe STOP mode, you can modify the parameter control DB with the PG's "Output Block" function. You must specify the parameter control DB when you initialize the system parameters with COM 95F. One big advantage of the control DB is that it allows you to define certain system parameters "during operation".

Before you process the input data in the user program, you must first subject it to plausibility and range checks in a so-called filter routine.
The filter routine must be approved just like any other user program block by the person conducting the acceptance test.

STOP per Programmer Request

> **Safety Note**
>
> There is no interlock on the switch from STOP to RUN via the programmer, and the switch can be initiated simply by pressing a button on the programmer. For this reason, you may not regard a programmer-induced STOP as safety condition.
> Always switch the S5-95F off for maintenance via its own switch.

Lengthening the Cycle Time via PG Functions

> **Note**
>
> The execution of PG functions increases the PLC cycle time. This applies particularly to the STATUS function, as the S5-95F processes the blocks in a special mode when this function is invoked. the blocks' runtimes are command-dependent, and are increased considerably in this mode. Interrupts will nonetheless be serviced.

Operating the Programmer in Safety, Quasi-Safety and Test Mode

Table 14-7. Operating the Programmer in Safety, Quasi-Safety and Test Mode

Functions Invoked on the Programmer	Abbr.	Test Mode		Quasi-/ Safety Mode	
		STOP	RUN	STOP	RUN
Block input and output					
Input block DB, FB, OB, PB, SB Parameter control DB	INPUT	 Yes Yes	 Yes Yes	 No Yes	 No No
Output block with editing DB, FB, OB, PB, SB Parameter-Bedien-DB	OUTPUT	 Yes Yes	 Yes Yes	 No Yes	 No No
Output block without editing DB, FB, OB, PB, SB	OUTPUT	 Yes	 Yes	 Yes	 Yes
Test					
Signal status display with editing FB, OB, PB, SB	STATUS	 Yes	 Yes	 No	 No
Signal status display without editing FB, OB, PB, SB	STATUS	 Yes	 Yes	 Yes	 Yes
PLC functions					
Start PLC	START	Yes	No	Yes	No
Stop PLC	STOP	No	Yes	No	Yes
Compress PLC memory	COMP	Yes	Yes	Yes	No
Status Variable	STAT VAR	Yes	Yes	Yes	Yes
Force Variable with editing	FORCE VAR	Yes	Yes	No	No
Force Variable without editing	FORCE VAR	Yes	Yes	Yes	Yes
PLC info					
Output address with editing	OUTP ADR	Yes	Yes	No	No
Output address without editing	OUTP ADR	Yes	Yes	Yes	Yes
Memory configuration	SPAUS	Yes	Yes	Yes	Yes
Read system parameters	SYSPAR	Yes	Yes	Yes	Yes
Output block stack	BSTACK	Yes	No	Yes	No
Output interrupt stack	ISTACK	Yes	No	Yes	No
Auxiliary functions					
Transfer block from PG PLC DB, FB, OB, PB, SB Parameter control DB	TRANS	 Yes Yes	 Yes Yes	 No Yes	 No No
Transfer block from PLC PG	TRANS	Yes	Yes	Yes	Yes
Overall PLC reset	RESET	Yes	No	Yes	No
Output directory	DIR	Yes	Yes	Yes	Yes

14.5 Measures for Securing the User Program Against Errors

Measures for Securing the User Program with the S5-95F

Before storing the user program (including DB1) on EPROM, it must be run on the S5-95F ***at least once in Test mode***. It is absolutely necessary that the S5-95F execute the user program because it computes a checksum each time the RESTART routine is run and enters it into DB1.

The S5-95F uses the checksum computed in test mode in its failsafe RESTART routine to check the user program. This allows the S5-95F to check for program integrity, which is a basic prerequisite for any failsafe system.

> **Note**
>
> The checksum via the user program can be computed in the S5-95F and with COM 95F beginning version V2.1 (see the COM 95F Manual). Do not confuse the checksum with the checksum for the test file generated with COM 95F.

Securing the User Program with the COM 95F Software Package

In addition to a purely functional test, you can use the COM 95F software package to test the user program for falsification due to programmer errors. COM 95F's conformance tool uncovers dangerous systematic faults/errors and hardware faialures in the programming device (see the COM 95F Manual).

14.6 Storing the User Program on Memory Submodules

For operation in safety mode, you must store your user program on EPROMs. Each basic unit must have a separate EPROM. The EPROMS have identical contents and are exactly the same.

The following table shows you how to store your user program on a memory submodule (EPROM).

Prerequisite:
Data blocks with variable data must have already been entered with COM 95F in the ”Operating System Parameters” screen form.

Table 14-8. Storing a User Program on EPROM

Step	Action
1	Load the user program in the S5-95F so that the S5-95F can compute the checksum (signature) via the user program.
2	Call the STEP 5 ”Block Transfer” package. Read out the tested user program from the S5-95F and store it in a file (see PG manual). Note that, when using older STEP 5 packages, some organization blocks have to be reloaded manually.
3	Call the STEP 5 ”Program EPROM” package. Copy the user program from the file to EPROM. Settings for programming the EPROM: • Mode: Byte • Checksum: No • Module number: See section 18.5 (see PG manual).
4	Perform comparison for changes and corruptions using the acceptance tool (see the COM 95F Manual) or subject the user program to a complete functional test.

Which Blocks Need to be Stored on the EPROM?

The following table shows which blocks you must store on the EPROM and which are optional.

Table 14-9. Blocks Which Must be Stored on the EPROM

Block ...	Compulsory/ Optional	Remarks
OB, PB, SB, FB and shells of the integral FBs	Compulsory	Organization blocks, program blocks, sequence blocks and function blocks make up the executable program.
DB 1	Compulsory	DB 1 contains the system parameters. You generate it with COM 95F.
DB with constant contents	Compulsory	Data blocks with constant contents must not change when the system is in operation. You must specify the numbers of the data blocks with constant contents with COM 95F.
DBwith variable contents	Optional	Data blocks with variable contents may be changed by the user program during operation. You must specify the numbers of the data blocks with variable contents with COM 95F. If you do not want to store these data blocks on the EPROM, you may also generate them in the user program.
Parameter control DB	Optional	You can change the parameter control DB by programmer input in safety STOP mode. You must set the number of the parameter control DB with COM 95F. If you do not want to store this data block on the EPROM, you may also generate it in the user program.
DB 252 DB 253	Optional	Data blocks DB 252 and DB 253 contain the Send and Receive mailboxes for failsafe data interchange. If these data blocks are not stored on the EPROM, they are generated automatically by the S5-95F.
DB 254	Optional	DB 254 contains the S5-95F system messages. If this data block is not stored on the EPROM, it is generated automatically by the S5-95F.

15 Error Diagnosis and Elimination

15.1 S5-95F Responses to Errors 15- 1
15.1.1 Hard STOP 15- 1
15.1.2 Soft STOP 15- 1
15.1.3 Passivation of a Signal Group 15- 1
15.1.4 Reaction in the User Program 15- 1
15.1.5 Error Indication 15- 2
15.1.6 System Event DB and OB37 . . . . . 15- 2

15.2 Error Indicator on the S5-95F Basic Unit 15- 2

15.3 System Event Data Block DB254 . . . . . 15- 3
15.3.1 System Identification and ID Number Entries 15- 3
15.3.2 Standard FB and Signature 15- 4
15.3.3 Information on System Responses 15- 4
15.3.4 Image of the Signal Groups 15- 5
15.3.5 Image of the Error Groups 15- 5
15.3.6 Static Image of I/O Errors 15- 6
15.3.7 Image of SINEC L1 Errors 15- 7
15.3.8 Error Stack Entries 15- 8
15.3.9 Evaluating the Error Block 15- 10

15.4 Acknowledging Errors and Deleting Entries in the System Event DB 15- 19

15.5 Printer Output of Error Messages Via a CP 521 SI 15- 20

15.6 Forwarding Error Messages to the SINEC L1 Master 15- 20

15.7 Evaluating Cycle Time Statistics 15- 21

15.8 Diagnostic Byte for Battery and Load Voltage 15- 22

Figures

15-1 Location of the Diagnostic LED 15- 2

Tables

15-1 Structure of the System Event DB 15- 3
15-2 Evaluating System Events in DB254 . . . . . 15- 11
15-3 Acknowledging System Events 15- 19
15-4 Memory Allocation for the Cycle Time Statistics 15- 21

15 Error Diagnosis and Elimination

This section covers the following:

- S5-95F responses to errors
- Error indicator on the S5-95F basic unit
- Structure of system event data block DB254
- Outputting error messages to printer via a CP 521
- Forwarding error messages to the master over the SINEC L1 bus
- Acknowledging errors
- Reading out cycle time statistics

15.1 S5-95F Responses to Errors

The S5-95F recognizes various types of errors, and responds to each in a different way. Possible error responses are:

- Hard STOP
- Soft STOP
- Passivation of a signal group
- Reaction from the user program
- Error message

The acknowledging of errors is discussed in detail in section 15.4.

15.1.1 Hard STOP

The S5-95F goes to a hard STOP when the failure of one of the subunits is such that failsafe scanning of the user program is no longer possible (for instance a hardware fault in the processor's internal RAM). A hard STOP always requires an overall system reset.

15.1.2 Soft STOP

The S5-95F is in a soft STOP when it can be switched to RUN without an overall system reset.

15.1.3 Passivation of a Signal Group

When the S5-95F detects an error at an input or output, it passivates this signal group (if you have initialized DB1 accordingly). The signal states of passivated DIs are read as "0"; DQs are set to "0", regardless of how they appear in the PIQ.

15.1.4 Reaction in the User Program

The signals of errored inputs can be logically combined if DB1 was initialized for user program response to a signal group failure. The S5-95F then either ANDs or ORs these inputs or provides the user with the last value to precede the error (old value).

15.1.5 Error Indication

An error is indicated when the S5-95F makes an entry in system event data block DB254 and the system responds accordingly (e.g. with STOP or passivation).

15.1.6 System Event DB and OB37

The S5-95F enters every error it detects into system event data block DB254. The reason for an entry might be an error or a system message. You should evaluate the entry in system event data block DB254 (with the COM 95F software or in your application program) and flag it as required. Each entry in system event data block DB254 invokes the error response OB (OB37).

15.2 Error Indicator on the S5-95F Basic Unit

The basic unit is equipped with a yellow error LED. The operating system sets this LED as soon as an error has been flagged in the system event DB. If the entry in the system event DB is not indicative of an error, e.g. "SINEC L1 o.k." or "SINEC L1 data path passivated", the operating system does not set the error LED.

The LED remains set

- In Test mode until the PLC once again goes from STOP to RUN
- In Safety and Quasi-Safety mode until the next STOP/POWER OFF/POWER ON/RUN sequence
- Until it is reset in the user program via FB255

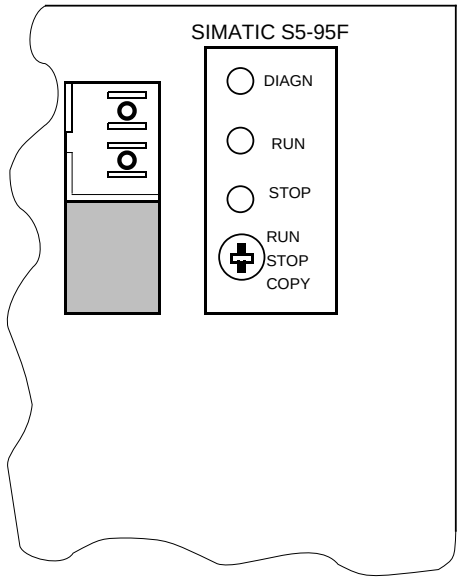


Figure 15-1. Location of the Diagnostic LED

Evaluate the system event data block (DB254) when the yellow error LED flags an error.

15.3 System Event Data Block DB254

The system event data block (DB254) is subdivided into several sections. These sections are shown in Table 15-1. Purpose, contents and characteristics of these sections are discussed in detail in the following sections.

Table 15-1. Structure of the System Event DB

Location in the DB	Sections of the System Event DB (DB254)
DW 0 to 1	Product identification and system ID number
DW 2 to 33	Standard FBs loaded and their actual signature
DW 34	Group info for responses to errors
DW 35 to 36	Image of the passivated signal groups
DW 37	Image of the error groups
DW 38 to 55	Image for errors at inputs and outputs
DW 56	Image for SINEC L1 errors
DW 57 to 60	Unassigned
DW 61 to 62	Error block counter and write pointer
DW 63	Unassigned
DW 64 to 191	Error entries with time stamp (a total of 16 error blocks of 8 data words each)

15.3.1 System Identification and ID Number Entries

Product Identification

Data word DW 0 in the system event DB of each basic unit contains the product identification code "095F$_H$". You must not change this code.

System ID Number

If your system comprises several S5-95Fs, each with its own user program, it is conceivable that the wrong EPROM submodule might inadvertantly be plugged into the wrong PLC. To prevent such a problem, a unique system ID number can be assigned to each S5-95F. The system ID number must be entered in DB1 when the system parameters are initialized.

In failsafe systems, the operating system reads the system ID number from DB1 and enters it into data word DW1 of the system event DB. From this point on, the S5-95F accepts only EPROM submodules with this system ID number. You can erase the ID number via a manual overall reset or via an overall reset initiated by the appropriate entry on the programmer.

Safety Note

When using the system ID number as security measure, remember that
- The system ID number may not be zero
- A manual overall reset and an overall reset initiated via the programmer sets the system ID number to zero.

15.3.2 Standard FB and Signature

You, as user, may never modify standard function blocks. In order to make sure that they remain unchanged, each standard FB has a signature.

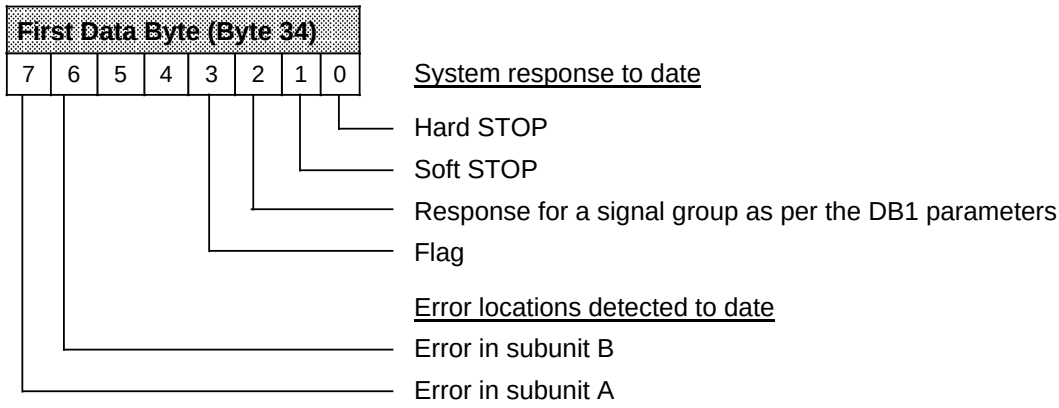
The S5-95F enters the signatures of the first 16 standard FBs loaded into data words DW 2 to DW 33 of DB 254. You can check the identity of these FBs on the basis of these entries.

Data Word	Loaded Standard FB
DW 2 DW 3	Block number of the first FB loaded Signature of the first FB loaded
DW 4 DW 5	Block number of the second FB loaded Signature
⋮ DW 32 DW 33	 Block number of the sixteenth FB loaded Signature

A manual overall reset or an overall reset initiated via the programmer deletes all entries in data words DW 2 to DW 33.

15.3.3 Information on System Responses

The operating system enters the system response variants which occurred following an overall reset in DR 34 (the low-order, or righthand, byte of data word DW 34). The entry corresponds to an ORing of "byte 0" of all error blocks (refer to section 15.3.9).



A manual overall reset or an overall reset initiated via the PG deletes the information in DR34.

15.3.4 Image of the Signal Groups

Each signal group is assigned one bit. The operating system sets the bit for a signal group as soon as an error was detected in that group.

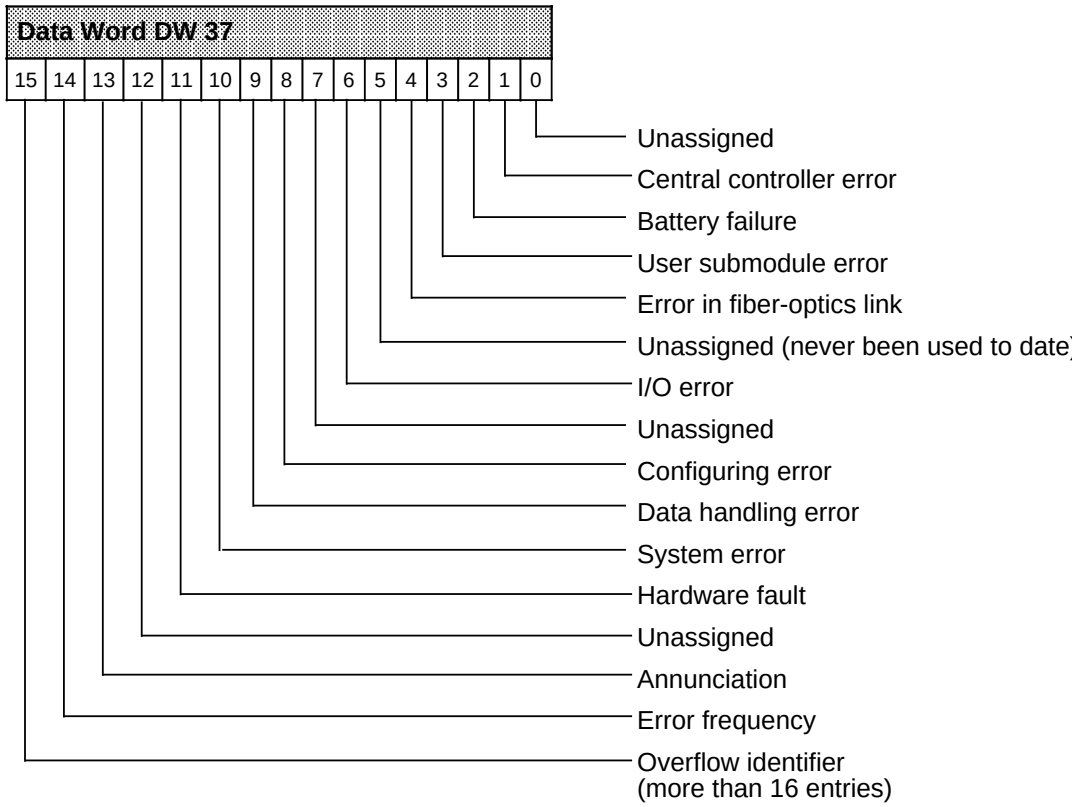
Bit in DW 35	35.15	35.14	35.13	35.12	35.11	35.10	35.9	35.8	35.7	35.6	35.5	35.4	35.3	35.2	35.1	35.0
Signal group	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16

Bit in DW 36	36.15	36.14	36.13	36.12	36.11	36.10	36.9	36.8	36.7	36.6	36.5	36.4	36.3	36.2	36.1	36.0
Signal group	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

You can delete the information in DW 35 and DW 36 by going from STOP to RUN, executing a manual overall reset, or initiating an overall reset via the programmer. If you depassivate a signal group with FB 255, the relevant bit is reset in the image of that signal group.

15.3.5 Image of the Error Groups

Each error group is assigned a bit. The operating system sets the bits. The bit set in the data word gives you an initial overview of the type of error involved.



You can delete the information in DW 37 via a manual overall reset or by initiating an overall reset on the programmer.

15.3.6 Static Image of I/O Errors

Each failsafe input and output is assigned a bit in one of data words DW 38 to DW 55. The operating system sets a bit in one of these data words when the associated I/O bit indicates an error.

You can delete the information in the static error image via a manual overall reset or by initiating an overall reset via the programmer.

Error Bits for Onboard DIs

Bit in DW 38	38.15	38.14	38.13	38.12	38.11	38.10	38.9	38.8	38.7	38.6	38.5	38.4	38.3	38.2	38.1	38.0
	Onboard DIs (byte 33)								Onboard DIs (byte 32)							
Error on channel	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0

Error Bits for Onboard DQs, Interrupt DIs and Onboard Counters

Bit in DW 39	39.15	39.14	39.13	39.12	39.11	39.10	39.9	39.8	39.7	39.6	39.5	39.4	39.3	39.2	39.1	39.0
	Onboard DQs (byte 32)								Counters				Interrupt DIs (byte 59)			
Error on channel	7	6	5	4	3	2	1	0	3	2	1	0			B	A

Error Bits for I/Os in Slots 0 and 1

Bit in DW 40	40.15	40.14	40.13	40.12	40.11	40.10	40.9	40.8	40.7	40.6	40.5	40.4	40.3	40.2	40.1	40.0
	DI/DQ byte 0								DI/DQ byte 1							
Error on channel	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0

Note the following:

- In the case of redundant digital modules, an error is always flagged only in the even-numbered byte.
- In the case of single-channel failsafe digital modules, an error is flagged in the even-numbered byte for subunit A and in the odd-numbered byte for subunit B.

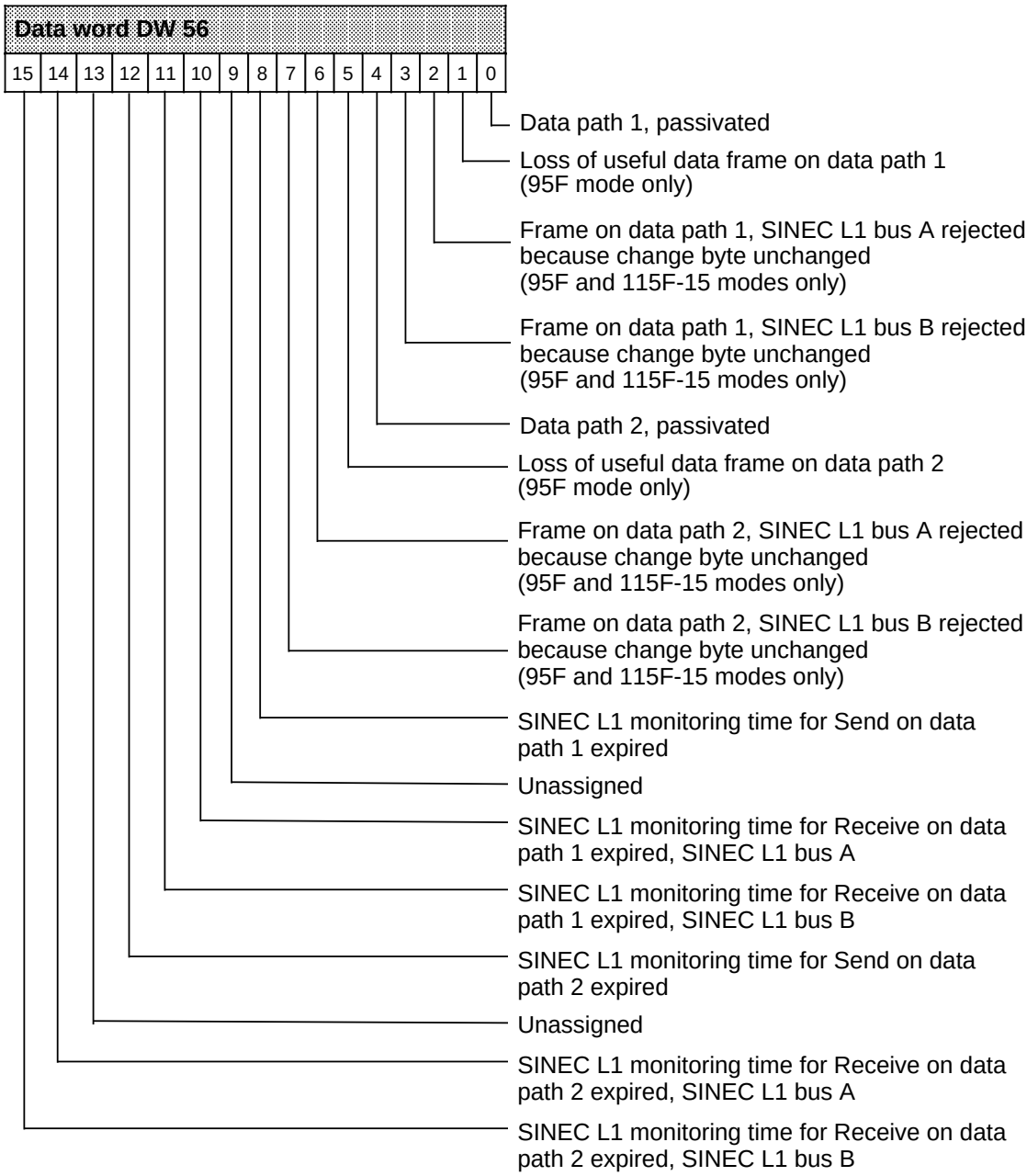
Error Bits for I/Os in Slots 2 and 3 to Slots 30 and 31

Data words DW 41 to DW 55 have the same meaning as DW 40, and are used to flag errors in the external I/Os in slots 2 and 3 to 30 and 31.

Data word DW 41 is assigned to slots 2 and 3, data word DW 42 to slots 4 and 5, data word DW 43 to slots 6 and 7, and so on.

15.3.7 Image of SINEC L1 Errors

Each SINEC L1 error is assigned a bit in data word DW 56.



The operating system updates the image for SINEC L1 errors every time a SINEC L1 error is flagged.
The system automatically resets the bits for frozen frames (bits 2, 3, 6, 7) when the error is no longer present.
You can delete the SINEC L1 error image by going from STOP to RUN, executing a manual overall reset, or initiating an overall reset via the programmer.

15.3.8 Error Stack Entries

All errors detected by the operating system during a cycle are flagged in the error stack. The error stack consists of:

- An error block counter
- A write pointer
- 16 error blocks of 8 data words each

Data Word	Contents of the Data Words
DW 61	Error block counter
DW 62	Write pointer
DW 63	Unassigned
DW 64 to 71	Error block 0 with bits for type of error, cause of error, auxiliary information and time stamp
DW 72 to 79	Error block 1 with bits for type of error, cause of error, auxiliary information and time stamp
DW 80	Error blocks 3 to 14
DW 184 to 191	Error block 15

You can delete the information in the error block by going from STOP to RUN, executing a manual overall reset, or initiating an overall reset via the programmer. Individual error flags can be reset with FB 255.

Error Block Counter

The operating system enters the number of errors which occurred in data word DW 61. The entry may be a number between 0 and 65535.

The error block counter is reset when a manual overall reset is executed or when an overall reset is initiated via the programmer.

Write Pointer

The write pointer identifies the last error block entered. This pointer is located in DL 62, and contains the number of the error block+1.

Error Blocks

Beginning in data word DW 64, a detailed report, including the cause of error, auxiliary information such as the I/O address or count, and the time stamp is entered for each error that occurs. Each such error report reserves 8 data words, and is sometimes referred to in the following as error block. The error blocks begin with data word n (n = 64, 72, 80, 88, ..., 184).

To facilitate understanding, the error block is subdivided into bytes which are numbered from 0 to 15.

Data Word	Error Block Contents			Bytes
n	Error location	System response	Main report	0 and 1
n+1	Auxiliary info 1			2 and 3
n+2	Auxiliary info 2			4 and 5
n+3	Auxiliary info 3			6 and 7
n+4	Internal data			8 and 9
n+5	Internal data			10 and 11
n+6	Day (BCD)		Month (BCD)	12 and 13
n+7	Hour (BCD)		Minute (BCD)	14 and 15

Time stamp

Evaluation of the error block is discussed in detail in the next section.

15.3.9 Evaluating the Error Block

You should evaluate the error block entry as soon as the yellow error LED on the base unit goes on.

You can evaluate the error in one of two ways:

- Direct reading out and interpreting of the system event DB
  or
- Using the COM 95F software (refer to the COM 95F manual)

If you have the COM 95F software package, COM 95F interprets the error blocks and provides a plaintext description of the error.

> **Note**
>
> Please note that some errors occur only in conjunction with other faults or errors which were duly reported as they occurred. For this reason, always evaluate all reported errors when doing a complete error/fault analysis.

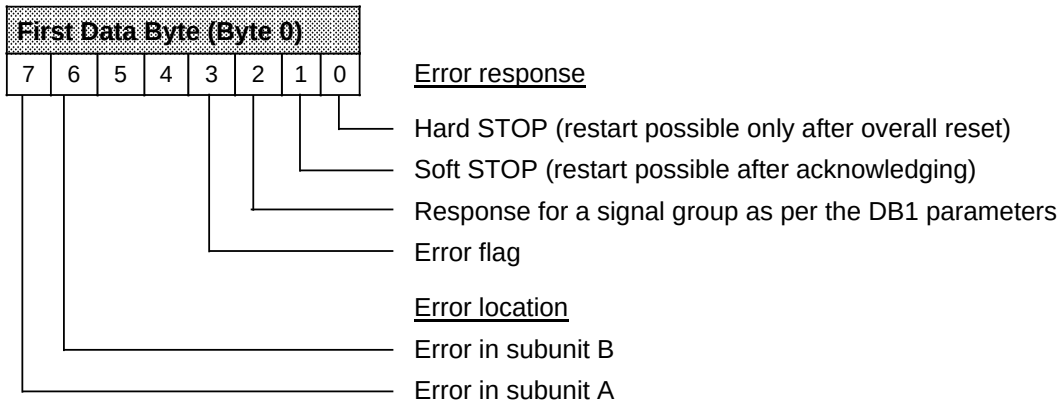
Direct evaluation of the error block is discussed below.

Display the system event DB on the programmer's monitor.

Read out DB254 from the PLC.
Ascertain the number of the last error block stored by viewing DR62.
The 1st DW in the error block is 64 + (error block number times 8).

Determine the error response and the subunit in which the error/fault occurred.

Evaluate byte 0. Determine the error response and the subunit in which the error/fault occurred. This information is coded in the first byte (byte 0) of the relevant error block.



Determine the cause of error.

Ascertain the identifier for the main error report by viewing the contents of byte 1, and locate the associated plaintext message in the table below.
If indicated in the table, also ascertain the auxiliary info by reading bytes 2 to 7.

Ascertain the time at which the error occurred.

View bytes 12 to 15 to find out the date and time when the error occurred (serves a practical purpose only when the integrated clock was properly set).

Overview of Error Reports

Table 15-2. Evaluating System Events in DB254

Error No. (Decimal)	Primary Information	Auxiliary Information	Corrective Measures
01	Hardware fault on the CPU board (Error response: Hard STOP or soft STOP)		Overall reset of basic unit Check supply voltage Replace basic unit
02	Hardware fault on the internal power supply board (Error response: Hard STOP)		Overall reset of basic unit Replace basic unit Note: S5-95F flags this error when the interrupt load is too high due to alarms/entries on programmer.
03	Subunit synchronization error (Error response: Soft STOP)		Check fiber-optics link Check switch setting for subunit ID
04	Too many interrupts (Error response: Soft STOP)		Shorten OB2 Reduce no. of INTs Replace basic unit
05	Hardware clock failed (Error response: Flag)		Replace basic unit
06	Basic units have different version numbers (Error response: Hard STOP)		Check version nos. Replace basic unit
07	Internal RAM error (hardware fault) (Error response: Hard STOP)		Replace basic unit

Table 15-2. Evaluating System Events in DB254 (continued)

Error No. (Decimal)	Primary Information	Auxiliary Information	Corrective Measures
10	Error in user program (error detected during program analysis) (Error response: Soft STOP)	Byte 2: 00_H: Illegal operation in user program 01_H: Timer or counter > 127 02_H: Nesting level exceeded 03_H: Illegal jump label 04_H: Illegal operation in interrupt routine (OB2) 05_H: Redundant module referenced with uneven address 06_H: Invalid STEP 5 parameter 07_H: 2nd operand in DO DW or DO FW unequal zero	Evaluate ISTACK Check program
11	Error in user program (error detected during program scan) (Error response: Soft STOP)	Byte 2: 30_H: Parameter error in FB call 31_H: Illegal operation in user program 32_H: Error in integral FB 34_H: Nesting level exceeded 35_H: Timer or counter > 127 36_H: DB not opened or no such DW in DB 37_H: Block nesting depth exceeded 38_H: Illegal reference in TIR statement 39_H: Illegal reference in LIR statement $3A_H$: Illegal reference in TNB statement	Check ISTACK Check program

Table 15-2. Evaluating System Events in DB254 (continued)

Error No. (Decimal)	Primary Information	Auxiliary Information	Corrective Measures
12	Error in user program (error detected during program scan) (Error response: Soft STOP)	Byte 2: 51_H: Substitution error in FB 52_H: Block nesting depth exceeded 53_H: DB not opened or no such DW in DB 54_H: DB generation error 55_H: Redundant module referenced with uneven address 56_H: G DB operation not allowed on DB0, 1 or 252 to 254, as these are reserved DBs 57_H: Non-existent block called 58_H: Illegal I/O access operation with T PY 59_H: Illegal I/O access operation with T PW $5A_H$: Illegal I/O access operation with L PY $5B_H$: Illegal I/O access operation with L PW $5C_H$: Interrupt caused exceeding of block nesting depth $5D_H$: Illegal reference in TIR statement $5E_H$: Illegal reference in LIR statement $5F_H$: Substitution error in integral block 60_H: DB for OB251 not opened 61_H: Illegal reference in TNB statement	Evaluate ISTACK Check program

Table 15-2. Evaluating System Events in DB254 (continued)

Error No. (Decimal)	Primary Information	Auxiliary Information	Corrective Measures
13	Error in user program (detected during program scan) (Error response: Soft STOP) (Error response: Error reported when byte 2 contains 71_{H})	Byte 2: 73_{H}: Illegal operation in user program 74_{H}: Range violation in DO DW/DI FW 75_{H}: OB2 or OB3 takes too long to execute 71_{H}: Interrupts disabled too long Byte 3: When *byte 2* contains 71_{H} or75_{H} 0: Cause in OB2 1: Cause in OB3 2: Error caused by IA/RA operation (see 8.2.8)	Evaluate ISTACK Check program
20	Memory submodule defective (Error response: Hard STOP)		Replace memory submodule
21	Subunits contain different memory submodules (Error response: Hard STOP)		Always use memory submodules with same Order No. in both subunits
22	Illegal memory submodule type (Error response: Hard STOP)		Use only correct memory submodules (Chapter 2)
23	Memory submodules have wrong system ID (Error response: Hard STOP)		Use submodules with right system ID
24	Memory submodules replaced (Error response: Hard STOP)		Never replace memory submodules while PLC is in RUN mode Insert memory submodule with different contents only after overall reset
25	Subunits have different user programs (Error response: Soft STOP)		Insert memory submodules with identical user programs
26	User program has incorrect signature (Error response: Soft STOP)		Give user program new signature Replace memory submodule Replace basic unit

* Operand in substitute statement illegal (T>127, EW>126, AW>126, MW=255 or OB 2 used)

Table 15-2. Evaluating System Events in DB254 (continued)

Error No. (Decimal)	Primary Information	Auxiliary Information	Corrective Measures
40	Hardware fault in onboard DI (Error response specified in DB1: Soft STOP or reaction in user program)	Byte 2: Byte no. Byte 3: Bit no.* Byte 5: Signal group	Check load voltage Replace basic unit
41	Hardware fault in onboard DQ (Error response specified in DB1: Soft STOP or reaction in user program)	Byte 2: Byte no. Byte 3: Bit no.* Byte 5: Signal group	Check load voltage (voltage including ripple 20 to 30 V) Check wiring Replace basic unit
42	Hardware fault on onboard hardware interrupts (Error response specified in DB1: Soft STOP or reaction in user program)	Byte 2: Byte no. Byte 3: Bit no.* Byte 5: Signal group	Check load voltage Replace basic unit
43	Hardware fault in onboard counter (Error response specified in DB1: Soft STOP or reaction in user program)	Byte 2: Byte no. Byte 3: Bit no.* Byte 5: Signal group	Check load voltage Replace basic unit
44	Hardware fault on external DI module (Error response specified in DB1: Soft STOP or reaction in user program)	Byte 2: Byte no. Byte 3: Bit no.* Byte 5: Signal group	Check load voltage Replace DI module
45	Hardware fault on external DQ module (Error response specified in DB1: Soft STOP or reaction in user program)	Byte 2: Byte no. Byte 3: Bit no.* Byte 5: Signal group	Check load voltage (voltage including ripple 20 to 30 V) Check module switch setting (P/M) Check wiring Replace module and/or bus unit

* FF_H or 255_D means that the bit number could not be ascertained

Table 15-2. Evaluating System Events in DB254 (continued)

Error No. (Decimal)	Primary Information	Auxiliary Information	Corrective Measures
48	Module incorrectly configured in DB1 (Error response specified in DB1: Soft STOP or reaction in user program)	Byte 2: Byte no. Byte 3: Bit no. Byte 5: Signal group	Check DB1 Check hardware
50	Onboard DI's discrepancy time too long (Error response specified in DB1: Soft STOP or reaction in user program)	Byte 2: Byte no. Byte 3: Bit no. Byte 5: Signal group	Check discrepancy time specified in DB1 Check wiring and sensor Replace basic unit
52	Onboard interrupt DI's discrepancy time too long (Error response specified in DB1: Soft STOP or reaction in user program)	Byte 2: Byte no. Byte 3: Bit no. Byte 5: Signal group	Check discrepancy time specified in DB1 Check wiring and sensor Avoid use of the "COMPRESS" function in RUN mode Replace basic unit
53	Onboard counter's discrepancy time too long (Error response specified in DB1: Soft STOP or reaction in user program)	Byte 2: Byte no. Byte 3: Bit no. Byte 5: Signal group	Check discrepancy time specified in DB1 Check wiring and sensor Replace basic unit Observe cut-off frequency
54	Exernal DI's discrepancy time too long (Error response specified in DB1: Soft STOP or reaction in user program)	Byte 2: Byte no. Byte 3: Bit no. Byte 5: Signal group	Check discrepancy time specified in DB1 Check wiring and sensor Replace module
57	Short circuit on sensor circuit to onboard DI (Error response specified in DB1: Soft STOP or reaction in user program)	Byte 2: Byte no. Byte 3: Bit no. Byte 5: Signal group	Check wiring
58	Short circuit on sensor circuit to onboard interrupt DI (Error response specified in DB1: Soft STOP or reaction in user program)	Byte 2: Byte no. Byte 3: Bit no. Byte 5: Signal group	Check wiring

Table 15-2. Evaluating System Events in DB254 (continued)

Error No. (Decimal)	Primary Information	Auxiliary Information	Corrective Measures
59	Short circuit on sensor circuit to external DI (Error response specified in DB1: Soft STOP or reaction in user program)	Byte 2: Byte no. Byte 3: Bit no. Byte 5: Signal group	Check wiring
60	SINEC L1 TIMER expired (Response: Error message)	Evaluate SINEC L1 error image mapped into DW 56 of DB254 (see 15.3.7)	Check SINEC L1 master/peer - Master/peer in RUN mode - Check connection/wiring Check SINEC L1 safety time • Increase call interval in polling list if necessary
61	SINEC L1 ok. (Response: Error message)	Evaluate SINEC L1 error image mapped into DW 56 of DB254 (see 15.3.7)	
62	SINEC L1 framing error (Response: Error message)	Evaluate SINEC L1 error image mapped into DW 56 of DB254 (see 15.3.7)	Check SINEC L1 master/peer - Master/peer in RUN mode - Check connection/wiring Check SINEC L1 safety time - Increase call interval in polling list if necessary Replace SINEC L1 peer's basic unit if frames frozen Ignore this message if it occurs during startup on the fault-tolerant SINEC L1 LAN.
63	SINEC L1 Send length invalid (Response: Soft STOP)	Byte 2: 01_H: Data path 1 02_H: Data path 2 Byte 3: Specified Send length	Select permissible Send length Send length must be in from 1 to 60 bytes
64	SINEC L1 data path depassivated (Response: Error message)	Byte 2: $0F_H$: Data path 1 depassivated $F0_H$: Data path 2 depassivated	Check wiring

Table 15-2. Evaluating System Events in DB254 (continued)

Error No. (Decimal)	Primary Information	Auxiliary Information	Corrective Measures
69	Configuring error in DB1 parameter block for short-circuit test (Response: Soft STOP)		Check configuring data in DB1 - Combining of interrupt DI and non-interrupt DI not allowed
70	Configuring error in DB1 (Error response: Soft STOP)	Byte 2: Syntax error in DB1, data word: (high address) Byte 3: Syntax error in data word: (low address)	Check configuring data in DB1
71	Error in standard FB (Error response: Soft STOP)	Byte 2: 88_H: FB in RAM $C8_H$: FB in EPROM Byte 3: Block No.	Reload standard FB
72	OB13 too long or specified OB13 interval too short (Error response: Soft STOP)		Optimize or shorten OB13 Increase OB13 call interval in DB1
73	Battery discharged Power-down in 72 h (Response: Error message)		Replace battery immediately
74	Battery failure exceeds 72 h (Error response: Soft STOP)		Insert battery and execute a cold restart
75	Load voltage for onboard I/Os failed or not connected (Response: Error message)		Check load voltage supply
76	Load voltage for counters and/or hardware interrupts failed (Response: Error message)		Check load voltage supply

15.4 Acknowledging Errors and Deleting Entries in the System Event DB

Acknowledge an error only after

- A failsafe response has been initiated
- The reported error has been rectified.

The S5-95F's response to an error or fault depends on its gravity. For this reason, there are several different ways to acknowledge an error, depending on

- the operating mode
  and
- the system response.

Acknowledging Errors

Table 15-3. Acknowledging System Events

System response	Acknowledged by	Comments
Hard STOP	Overall reset - via programmer - manual (see 2.5.4)	All error flags in DB254 are reset
Soft STOP	RUN/STOP switch on both subunits to STOP, then RUN/STOP switch on both subunits to RUN (RUN via programmer not possible)	All error flags in DB254 are reset
Passivation of I/Os	Starting depassivation block FB255 with job number 1,x (see Chapter 9)	Error flags in DB254 belonging to signal group x are reset
Standard value formation (AND, OR, old value) for deviating DI signals ***and*** setting of flag in DB254 ***and*** subsequent response at the user level	Starting depassivation block FB255 with job number 0,0 (see Chapter 9)	All error flags in DB254 are reset
Error report (state of load voltage and backup battery	Starting depassivation block FB255 with job number 0,0 (see Chapter 9)	All error flags in DB254 are reset

15.5 Printer Output of Error Messages Via a CP 521 SI

If you wish, you can output S5-95F error messages to printer. To do so, you need a CP 521 SI and two data blocks supplied with the COM 95F software. These blocks are on file CP521DST.S5D. Note that blocks DB1 and DB10 are for the CP 521 SI, and may not be loaded into the S5-95F.

DB1 contains the parameters for the printer. These parameters are pre-initialized for a printer with a V.24 (RS232C) interface as follows:

- Data signalling rate 9600 baud
- Parity Arbitrary
- BUSY signal Yes
- Data format 7 data bits with parity
- Hardware handshake No

You must edit DB1 if you are using a printer with other features. You will find a detailed description in the CP 521 SI Manual.

DB10 contains the message texts to appear on the printer. These texts are shorter than those used in the COM 95F software, and consist of error number, primary information and auxiliary information (see section 15.3.9). For corrective measures, please refer to the Manual (see Table 15-2).

When initializing the system parameters with COM 95F, you must initialize the CP 521 SI as signalling module (see COM 95F Manual).

15.6 Forwarding Error Messages to the SINEC L1 Master

The S5-95F can automatically forward system messages to the SINEC L1 master of bus B, thus providing you with a central point for evaluating, displaying and logging all messages.

When DB1 is initialized accordingly (see COM 95F Manual), the S5-95F forwards each error flagged in DB254's error stack to the SINEC L1 master of bus B. Non-failsafe frames to the SINEC L1 master have a fixed length of 16 net data bytes. The SINEC L1 master can identify frames as system messages on the basis of the source slave number and the frame length. (For this reason, never send user frames with a length of 16 net data bytes from the S5-95F to the bus master).

15.7 Evaluating Cycle Time Statistics

The S5-95F collects statistical data on cycle times. This statistical data is located in internal RAM in the address space extending from 8004_H to $80C3_H$. You can evaluate the cycle time statistic by invoking the programmer function OUTPUT ADDRESS.

Functionality

The S5-95F computes the time needed for each cycle and assigns it to an interval. A counter is then assigned to each interval. The S5-95F increments a counter by one whenever the cycle time being measured belongs to that counter's interval.

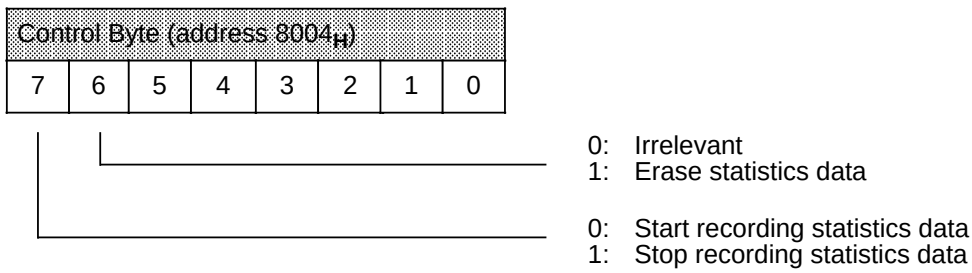
The statistics function is located in internal RAM and comprises

- 95 counters of 2 bytes each (see Table 15-4) and
- a control byte

Table 15-4. Memory Allocation for the Cycle Time Statistics

Address	Description	
8004_H	Control byte	
8005_H	Unassigned	
8006_H	Counter 1, low-order byte	For cycles with a time < 60 ms
8007_H	Counter 1, high-order byte	
8008_H	Counter 2, low-order byte	For cycles with 60 ms cycle time < 70 ms
8009_H	Counter 2, high-order byte	
. . .	. . .	. . .
$80C2_H$	Counter 95, low-order byte	For cycles with 990 ms cycle time < 1000 ms
$80C3_H$	Counter 95, high-order byte	

Contents of the Control Byte



Erasing Cycle Time Statistics Data

Cycle time statistics data is erased

- by an overall reset
- by setting bit 6 in the control byte

Starting the Recording of Cycle Time Statistics Data

The recording of cycle time statistics data is started or resumed

- when the S5-95F is set to RUN
- by resetting bit 7 in the control byte with the programmer's "Output Address" function

Stopping the Recording of Cycle Time Statistics Data

The recording of cycle time statistics data is stopped

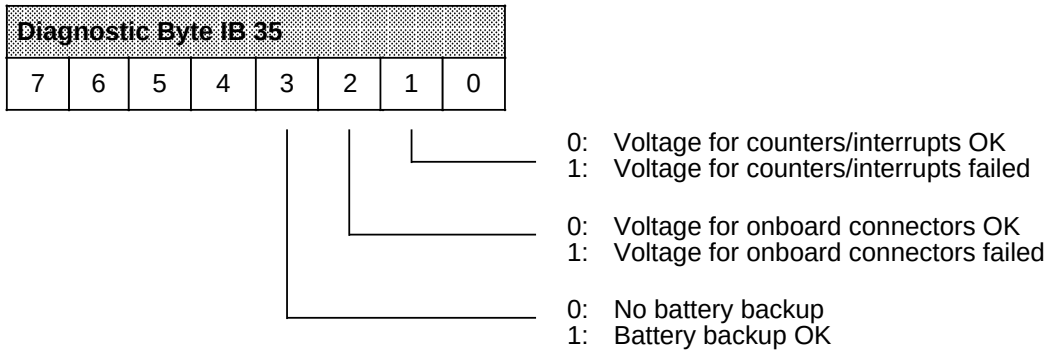
- when a counter reaches its maximum value of $FF00_H$
- by setting bit 7 in the control byte with the programmer's "Output Address" function

> **Note**
>
> In safety and quasi-safety mode, you cannot control the cycle time statistic when the S5-95F is at RUN. The operating system starts the recording of cycle time statistics data in its RESTART routine, and the function remains active.

15.8 Diagnostic Byte for Battery and Load Voltage

You can evaluate the following by querying diagnostic byte IB 35:

- The state of the backup battery
- The state of the load voltage for onboard DIs/DQs
- The state of the load voltage for counters and interrupt inputs



16 High-Availability Failsafe Control with S5-95F

16.1 Schematic Wiring Diagram of the Data Link 16- 2

16.2 Data Interchange for the High-Availability S5-95F -FB 230- 16- 4
16.2.1 Block Parameters for FB 230 . . . . . 16- 4
16.2.2 Description of the Parameters for FB 230 . . . . . 16- 5
16.2.3 Operands Reserved by FB 230 . . . . . 16- 10
16.2.4 How FB 230 Works 16- 11
16.2.5 Operating States and Transitional States of the High-Availability S5-95F 16- 14
16.2.6 Time Response of the FB 230 . . . . . 16- 16

16.3 Initializing Parameters with COM 95F 16- 17

16.4 What to Watch for When Writing User Programs 16- 17
16.4.1 Structure of a Program for the High-availability S5-95F 16- 18

16.5 Timing in the User Program 16- 20

16.6 Using Analog Inputs in the H/F System 16- 21

16.7 Using SINEC L1 in the H/F System 16- 21

16.8 Sample Program for Initializing FB 230 . . . . . 16- 22

16.9 Connecting I/Os to High-Availability S5-95Fs 16- 27
16.9.1 Connecting Digital Inputs 16- 27
16.9.2 Connecting Digital Outputs 16- 30
16.9.3 Connecting Analog Inputs 16- 34

16.10 Timing for H/F Systems -FB 231 and FB 238- 16- 36
16.10.1 Block Parameters for FB 231 . . . . . 16- 36
16.10.2 Block Parameters for FB 238 . . . . . 16- 39

Figures

16-1 Basic Hardware Configuration for a High-Availability S5-95F 16- 1
16-2 Schematic Wiring Diagram of the Complete Data Link 16- 2
16-3 Sample Annunciator Panel 16- 3
16-4 Structure of a Program for an H/F System 16- 18
16-5 Connecting Failsafe Digital Inputs with One Failsafe Sensor 16- 27
16-6 Connecting Failsafe Digital Inputs with Two Sensors 16- 28
16-7 Connecting a Non-Failsafe Digital Input 16- 29
16-8 Direct Actuator Control in a P/M Circuit 16- 30
16-9 Indirect Actuator Control in a P/M Circuit 16- 31
16-10 Indirect Actuator Control in a P/P Circuit 16- 32
16-11 Direct Actuator Control for Non-Failsafe DQs 16- 33
16-12 Connecting Failsafe Analog Inputs with Double Discrepancy Evaluation 16- 34
16-13 Connecting Non-Failsafe Analog Inputs 16- 35
16-14 Signal Time Diagram 16- 41

Tables

16-1 FB 230 Parameters 16- 5
16-2 Remarks Regarding Individual Block Parameters 16- 8
16-3 Reserved Operands 16- 10
16-4 Parameters for Cyclic Data Transfer 16- 22
16-5 Parameters for Update Mode 16- 23
16-6 I/O Types for Digital Input Modules in H/F Systems 16- 27
16-7 I/O Types for Digital Outputs in H/F Systems 16- 30
16-8 I/O Types for Analog Input Modules In H/F Systems 16- 34
16-9 Block Parameters for FB 231 . . . . . 16- 37
16-10 Operands Reserved by FB 231 . . . . . 16- 37
16-11 Structure of the Data Block for Timers 16- 38
16-12 Block Parameters for FB 238 . . . . . 16- 40

16 High-Availability Failsafe Control with S5-95F

High-availability failsafe control systems are required more and more frequently in process engineering, as even the shortest downtimes often result in high production losses.

The S5-95F can help you build a high-availability failsafe control system, for the principle is extremely simple. All you have to do is

- connect two failsafe S5-95Fs to one another via a DI/DQ link and
- call and initialize a function block integrated in the S5-95F in your user program.

High-Availability S5-95F

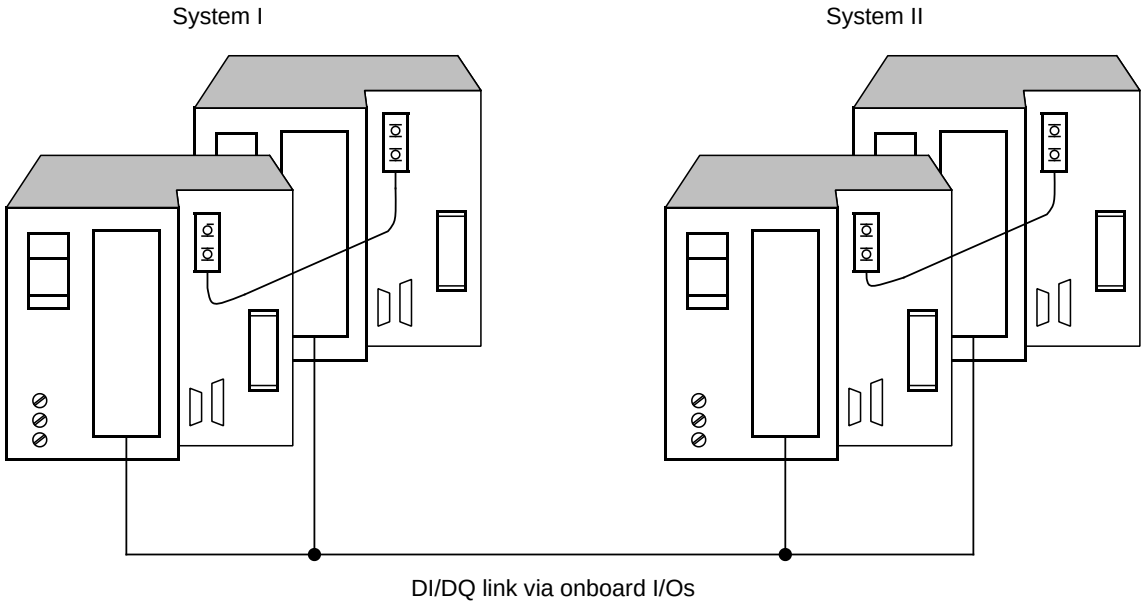


Figure 16-1. Basic Hardware Configuration for a High-Availability S5-95F

Prerequisites

A number of hardware and software prerequisites must be fulfilled in order to implement the high-availability S5-95F.
In addition to the DI/DQ link, the two systems must have identical hardware.
The same user program must be present in all subunits. All EPROMs must show the same checksum. All of these prerequisites are checked by the person or persons conducting the acceptance test.

A number of exceptions and restrictions must be observed when writing the user program (see section 16.4).

16.1 Schematic Wiring Diagram of the Data Link

Figure 16.2 shows the wiring of the DI/DQ link for the two failsafe systems in schematic form.

> **Safety Note**
>
> Correct wiring of the DI/DQ link is a basic prerequisite for error-free operation. For this reason, it must be checked by the inspector conducting the acceptance test.

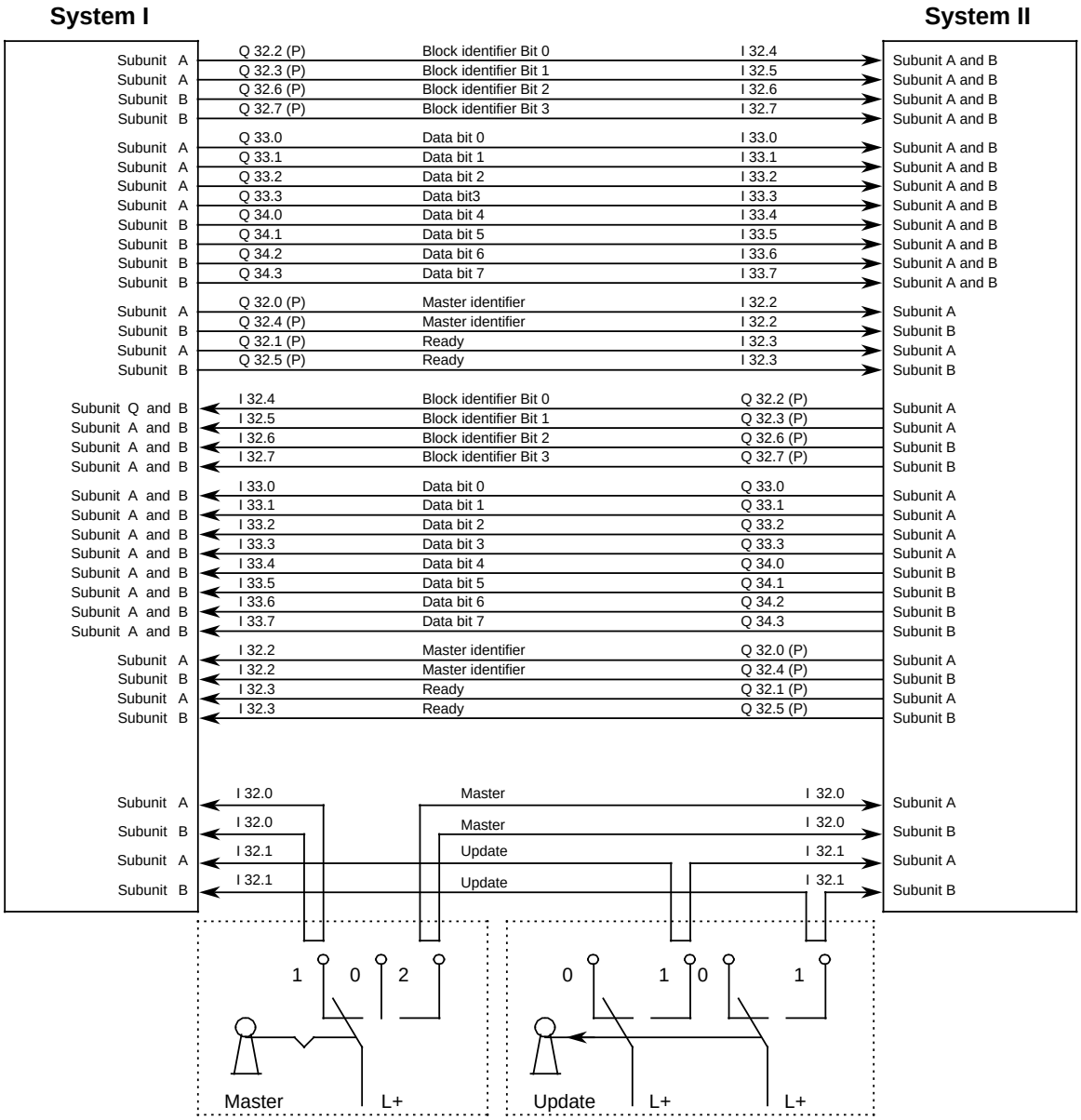


Figure 16-2. Schematic Wiring Diagram of the Complete Data Link

Remarks Regarding Figure 16.3

For each of the two failsafe systems, CPU and onboard I/Os must be supplied by the same 24 V DC source (otherwise the system will go to STOP with a discrepancy error in the event of a power outage).

Wiring of the ”Master” keylock switch must be redundant from the switch to the S5-95F. As throw switch, the ”Master” switch must have two active positions and one neutral position.
The selection as to which unit is to control the process (master) is made with the ”Master” keylock switch (effective for either system I or system II).

Wiring of the ”Update” switch must also be redundant.
The key-operated pushbutton must be used with two independent positive-action NO contacts.
The master system updates the standby system via the ”Update” keylock switch.
This key-operated pushbutton affects both systems simultaneously.

Sample Annunciator Panel with Switches and Indicators for the H/F System

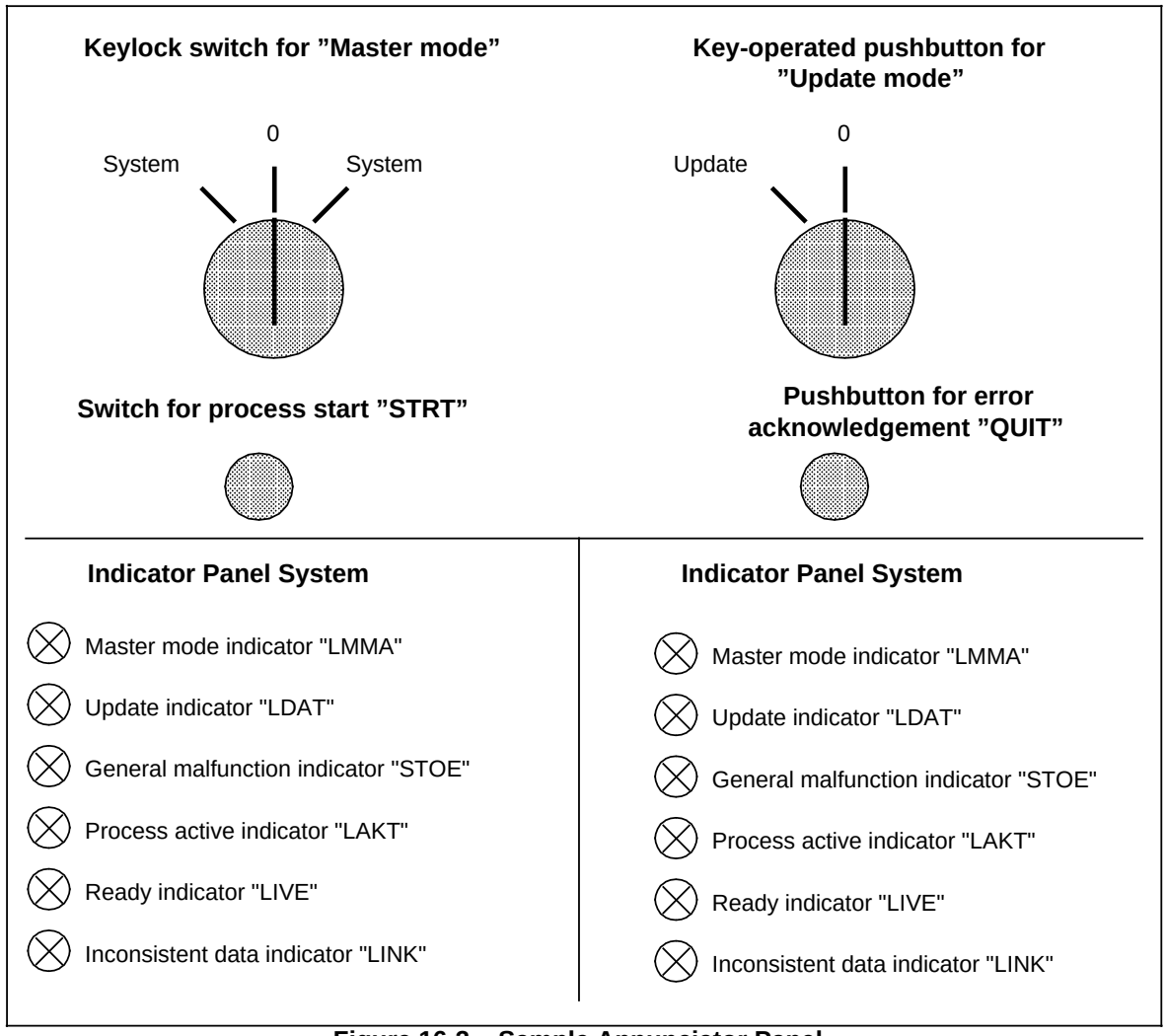


Figure 16-3. Sample Annunciator Panel

16.2 Data Interchange for a High-Availability S5-95F -FB 230-

Function block FB 230 ensures the safe interchange of data between the linked systems so that, should one system fail, the other can continue controlling the process.

Functions of FB 230

FB 230 executes the following functions:

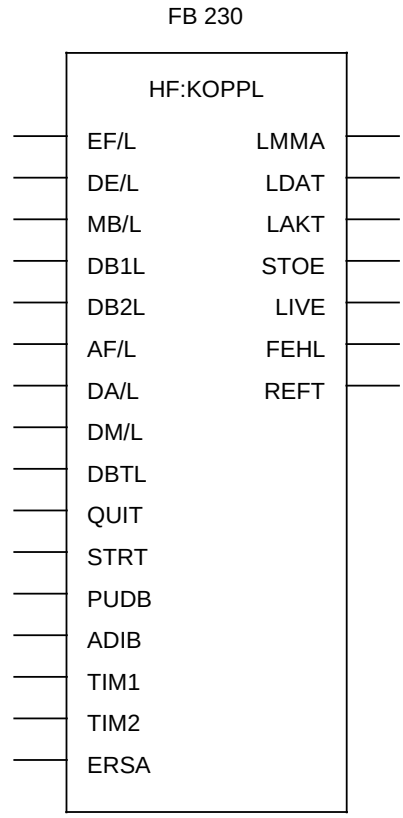
- Control of the master's operational performance.
- Control of the standby system's operational performance.
- Updating of the standby system by the master during the startup routine.
- Synchronization of both systems (master and standby) in cyclic operation.
- Generation of a time base for synchronous timing.
- Monitoring of the data link.
- Generation of operational flags and fault flags for the H link.

16.2.1 Block Parameters for FB 230

STL

```
0000        :
0001        :
0002        :JU FB 230
0003  NAME   :HF:KOPPL
0004  EF/L   :
0005  DE/L   :
0006  MB/L   :
0007  DB1L   :
0008  DB2L   :
0009  AF/L   :
000A  DA/L   :
000B  DM/L   :
000C  DBTL   :
000D  LMMA   :
000E  LDAT   :
000F  LAKT   :
0010  STOE   :
0011  LIVE   :
0012  LINK   :
0013  FEHL   :
0014  QUIT   :
0015  STRT   :
0016  PUDB   :
0017  ADIB   :
0018  REFT   :
0019  TIM1   :
001A  TIM2   :
001B        :    BEC
001C        :    ***
```

CSF



16.2.2 Description of the Parameters for FB 230

Table 16-1. FB 230 Parameters

Parameter	Description	Type		Contents	Remarks
EF/L	Start/length of area for failsafe DIs to be transferred cyclically	D	KY	KY=x,y x= Start address of the failsafe DIs y= Number of reserved slots	Max. possible address range: Addresses 0 to 31 Max. 16 failsafe DI bytes possible
DE/L	Start/length of area for non-failsafe DIs to be transferred cyclically	D	KY	KY=x,y x= Start address of the non-failsafe DIs y= Number of reserved slots	Max. possible address range: addresses 0 to 31 Max. 32 non-failsafe DI bytes possible
MB/L	Start/length of area for flags to be transferred cyclically	D	KY	KY=x,y x= Start address of flag byte y= Number of flag bytes	Max. possible address range: addresses 0 to 199 Max. 200 bytes
DB1L	Block no./length; user DB1 for updating	D	KY	KY=x,y x= Data block number y= Number of data words	Possible block numbers: DB2 to DB250 Max. 240 DW in length
DB2L	DB no./length; user DB2 for updating	D	KY	KY=x,y x= Data block number y= Number of data words	Possible block numbers: DB2 to DB250 Max. 240 DW in length
AF/L	Start/length of area for updating failsafe DQs	D	KY	KY=x,y x= Start address of the failsafe DQs y= Number of reserved slots	Max. possible address range: Addresses 0 to 31 Max. 16 failsafe DQ bytes possible
DA/L	Start/length of area for updating non-failsafe DQs	D	KY	KY=x,y x= Start address of non-failsafe DQs y= Number of reserved slots	Max. possible address range: Addresses 0 to 31 Max. 32 non-failsafe DQ bytes possible
DM/L	Start/length of area for flags for updating	D	KY	KY=x,y x= Start address of flag byte y= Number of flag bytes	Max. possible address range: Addresses 0 to 199 Max. 200 bytes
DBTL	DB no./length; timer DB	D	KY	KY=x,y x= Data block number y= 70	Possible block numbers: DB2 to DB250

Table 16-1. FB 230 Parameters (continued)

Parameter	Description	Type		Contents	Remarks
LMMA	Indicator for ”Master” mode	Q	BI	Non-failsafe DQ	DQ must not lie in same area as cyclic and update DQs
LDAT	Indicator for ”Update” mode	Q	BI	Non-failsafe DQ	DQ must not lie in same area as cyclic and update DQs
LAKT	Indicator for ”Active Process Control” mode	Q	BI	Non-failsafe DQ	DQ must not lie in same area as cyclic and update DQs
STOE	”General Fault” indicator	Q	BI	Non-failsafe DQ	DQ must not lie in same area as cyclic and update DQs
LIVE	Indicator for ”Data Transfer Error”	Q	BI	Non-failsafe DQ	DQ must not lie in same area as cyclic and update DQs
LINK	Indicator for ”Inconsistent Data” A	BI		Non-failsafe DQ	DQ must not lie in same area as cyclic and update DQs
FEHL	Internal diagnostic error flag	Q	W	FW0 to FW 198	Bit assignments 0=Error, master -> Standby log. 1 1=Error, master -> Standby log. 0 2=Error, standby -> Master log. 1 3=Error, standby -> Master log. 0 4=Error, data receive, master 5=Error, data trans., master 6=Error, ready, master 7=Error, ready, standby standby 8=Error, master failed 9=Error, useful data comp. master 10=Reset DQs after error 11=Error, updating 12=Error, parameters not OK 13=Error, standby not updated 14=Unassigned 15=Unassigned
QUIT	Input for fault acknowledgement	I	BI	Single-channel input I 0.0 to I 31.7	
STRT	Input for starting the master in stand-alone mode	I	BI	Single-channel input I 0.0 to I 31.7	
PUDB	Internal buffer DB for Send/Receive	B		DB 2 to DB 250	Required block length 256 DW

Table 16-1. FB 230 Parameters (continued)

Parameter	Description	Type		Contents	Remarks
ADIB	Updating in RUN mode	Q	KF	KF=1 Updating in RUN mode possible KF=0 Updating in RUN mode not possible	When ADIB = 0, updating of the standby system is possible only after a cold restart has been executed for the master.
REFT	Reference timer location for FB 231	Q	W	FW 0 to FW198	
TIM1	Data link timeout	D	KF	KF=80	Value is set one time only and adapted only in the event of a new PLC output.
TIM2	Data link timeout	D	KF	KF=800	Value is set one time only and adapted only in the event of a new PLC output.

Remarks Regarding Individual Block Parameters

Table 16-2. Remarks Regarding Individual Block Parameters

Parameter	Additional Remarks
EF/L	The EF/L parameter must always specify all failsafe inputs used in the control program. All failsafe inputs must be located in one contiguous block.
DE/L	The DE/L parameter must always specify all non-failsafe inputs used in the control program. All non-failsafe inputs must be located in one contiguous block.
EF/L,DE/L,AF/L, DA/L,MB/L, DM/L	The specified area may not overlap with other areas or the onboard I/Os.
EF/L,DE/L,AF/L, DA/L	As length parameters, these parameters must always specify the number of modules of the corresponding type used in an F system. In the case of failsafe DIs/DQs, note that there are two modules per I/O byte.
MB/L	The flags to be specified by the MB/L parameter are used to synchronize asynchronous events. These flags must be located in one contiguous block. 0.0 may be specified if there are no asynchronous events to be synchronized. In any case, this area should be kept as small as possible in order to keep the cycle time requirement for cyclic synchronization to a minimum. As a rule, two to six bytes are sufficient for many applications. Recommendation: Reserve flag bytes beginning flag byte FY4.
MB/L,DM/L	The flag areas defined by parameters MB/L and DM/L may not overlap. The auxiliary flags initialized in FB 230 may not be located in these areas. The individual flag areas may not overlap with temporary flags FY 200 to FY 255.
DB1L,DB2L	The data blocks to be initialized in parameters DB1L and DB2L are the only data blocks which may be used in the control program to store data. (Exception: Blocks for preprocessing asynchronous events in area 4 at the end of OB1). These data blocks may not exceed 240 data words. If no data blocks are used, 0,0 may be specified as parameter value. If the data blocks are not needed in their full length, the length parameter should be reduced accordingly to save updating time.
AF/L	The AF/L parameter must always specify all failsafe outputs used in the control program. All failsafe outputs must be located in one contiguous block.
DA/L	The DA/L parameter must always specify all non-failsafe outputs used in the control program. All non-failsafe outputs must be located in one contiguous block. Exception: The output byte for showing the operational states of FB 230 must lie outside this area.
DM/L	The DM/L parameter must specify all flags used in the control program (with the exception of the area already entered as MB/L parameter). The flags must be located in one contiguous block. In order to keep the updating time to a minimum, the unused flag area should be excluded.

Table 16-2. Remarks Regarding Individual Block Parameters (continued)

Parameter	Additional Remarks
DBTL	The block number of the specified timer DB must be identical to the TDB parameter specification in FB 231 (cyclic time processing). Only the complete DB can be transferred. The timer DB has a fixed length of 70 data words. If no timers are needed in the control program, the parameter can be preset with 0.0. Use of the operating system's timers is not allowed in an H system. Recommendation: Use DB 231.
LMMA,LDAT,LAKT,STOE,LIVE,LINK	These parameters are provided for controlling lamps in the control cabinet. An output byte on a non-failsafe output module must be reserved for this purpose. The remaining output bits on that same module must remain unassigned, and may not be used for another purpose. The output byte may not be located in the same areas as the values defined by the DA/L and AF/L parameters. The output byte is also not set when the process output image is reset due to an error. As a rule, the states of the bits on the two F systems differ. The bits may therefore not be logically gated in the control program. Recommendation: Use the first output byte in the non-failsafe output area.
FEHL	The error bits are acknowledged in the restart routine or with the Acknowledge button, and represent an additional diagnostic aid. They may differ in the two F systems (master and standby). If one system fails, follow-up annunciations appear in the other system. The flag word initialized here may not be located in the same areas as the DM/L and MB/L parameters. The bits may not be logically gated in the control program. Recommendation: Use FW2.
PUDB	This DB serves as auxiliary DB for FB230. The user must define it as a DB comprising 256 data words, and preset it to KH0000. This data block may not be automatically generated in one of the restart branches. Recommendation: Use DB 230.
ADIB	The contents of this parameter are determined by the inspector responsible for the system acceptance test. This parameter must be set to 0 in systems in which update monitoring is not possible.
REFT	This parameter must be identical to the REFT parameter in FB 231. The flag word initialized here must not be located in the same areas as the DM/L and MB/L parameters. Recommendation: Use flag word 0.

16.2.3 Operands Reserved by FB 230

Table 16-3. Reserved Operands

Type	Reserved Area
Temporary flags	FY 200 to FY 255
Timers	T0

The execution times for FB 230 can be found in section 16.2.6.

16.2.4 How FB 230 Works

Initial State and Update Procedure

Start Procedure:

Both failsafe systems have the same FB230 and the same user program.
The DI/DQ data link is wired as per the wiring diagram.
The front connectors are pins for L + and ground potential.
The master keylock switch and the update key-operated pushbutton are in the neutral position (initial state).

Switch on all four units via the On/Off switch. Set the mode switch to RUN.
Both F systems assume a passive wait state, that is to say, the control program is not processed in either system.

In the initial state, the Update indicator (LDAT) lights up on both systems. This indicator shows that neither system has as yet been updated.

Assign the master keylock switch to one system (for example system I / DI32.0).

An indicator lamp (LMMA) shows the operator which system is in the Master mode. The function block in this system recognizes that it is in Master mode and informs the standby system of this fact via DQs (DQ 32.0 / 32.4).

Update Procedure:

Updating is required the first time the system is put into operation or when a previously powered-down F system is once again put into operation.
When the Update key-operated pushbutton (which must remain actuated during updating) is actuated, the wiring system test for the DQ/DI link is tested before both systems are updated. As long as the update indicator (LDAT) is on, the master is updating the standby system. Updating includes all failsafe DIs, non-failsafe DIs, flags, failsafe DQs, non-failsafe DQs, the data block for timers (DBTL), and the data blocks for user timers (DB1L and DB2L). If the update procedure terminates without error, the update indicator goes out on both systems.
The control programs in master and standby are not actively processed during updating. The I/Os are not updated.
If the update procedure is aborted, the LDAT indicator as well as the fault indicator light up. A diagnostic bit is set in the error word.

If the standby cannot be updated, it is possible to put the master into operation with the STRT pushbutton. The master then controls the process as stand-alone system.

Once the standby system is ready to be put back into operation, the update procedure can be initiated by pressing the ”Update” key-operated pushbutton a second time.
The process is then controlled automatically once the two controllers have been synchronized. If the key-operated pushbutton is released too soon, the update procedure is aborted and the process controlled by the former master controller.

In situations where updating in this form is not possible for reasons of safety, the process must be halted and the controller that is to do the updating must be switched off and then restarted.

> **Warning**
>
> When updating takes place during operation, the master does not control the process while the update procedure is in progress.
> The last process output image is frozen for the duration of the update procedure.
> This function may be initiated only when there is no danger to the controlled plant (in a steady-state condition, for instance after a process was started, during a heating-up phase, etc.).
> The use of this mode is allowed only when the plant is under the supervision of experienced personnel. If the inspector does not permit use of this mode, the ADIB parameter must be preset to ”0”.

If the updating procedure was aborted manually or because of an error, the standby system may show inconsistent data.
In this case, the LINK indicator on the standby system goes on.
In this situation, there is an interlock on the standby system. This also applies when the standby system is to control the process in stand-alone mode should the master controller fail. The interlock can be defeated only by a new, successful update via the master or after executing a general reset and reloading the standby.

The LINK indicator (inconsistent data) on the standby unit stays on during the entire update procedure.
This is not an error, but only shows that the standby controller cannot be started in this state should the update procedure be aborted.
The LINK indicator goes out when the update is successfully completed.

OB31 calls are interspersed throughout FB 230 at the data exchange points (cycle time triggering). The value set in DB1 with COM 95F is thus decisive here for cycle time monitoring.
The value to be set here depends on the amount of data to be transferred during the update procedure (see Time Response of FB 230).

Cyclic Operation

During normal operation, F systems are synchronized at the beginning of the OB1 cycle.
When both systems have been synchronized, the DI/DQ link is tested; then the time base, all of the input signals used in the program (failsafe / non-failssafe DIs), and a limited number of user-specific flag bytes are transferred. The amount of data to be transferred and its location are specified via block parameters.
An image of this data is then generated in the relevant data areas. The standby then works in these areas with the master’s data.

Once the data interchange has been completed, the control program in OB1 is processed in both F systems.
The control programs in the two F systems now have the same program state, thus resulting in the same states for the process output image at the end of the OB1 cycle.
The process images of both F systems are output to the I/Os at the end of the OB1 cycle.

Cyclic Data Transfer at the Beginning of OB1

At the beginning of the OB1 cycle, all inputs and a number of synchronization flags are transferred from the master to the standby (parameters EF/L, DE/L, MB/L).
Note that the data transfer requires a not inconsiderable amount of time, thus increasing the program cycle time.

Because the cycle time needed by function block FB 230 is determined primarily by the number of input bytes, time-critical systems can be configured with only a limited number of inputs (also see 'Time Response of FB 230').

Performance When Errors Occur During Signal Transfer

Special procedures for data security and error detection are implemented during data interchange between the two F systems.

In each PLC cycle, the data link is tested for wirebreaks and short-circuits. Data transfer errors caused by short-term interference are automatically detected and corrected.

If there is persistent interference or a defect in the data link, the link is taken out of operation.
The standby system is then switched to a passive state in which the control program is no longer processed, and in which all outputs are reset.
This state must be acknowledged.

At the same time, the master goes to stand-alone mode.
In this mode, there are no attempts at synchronization, nor are there any attempts at data interchange with the standby system. The master alone controls the process.
This state is not exited until it has been acknowledged and the standby has been successfully updated.

Proper functioning of the monitoring mechanisms requires correct wiring of the data link.

16.2.5 Operating States and Transitional States of the High-Availability S5-95F

Switching On the Operational H/F System

Before the system can be switched on, the same program module (same checksum) must be plugged in and the required data blocks loaded into all controllers.

When a previously active H/F system is switched on, both F systems are first set to RUN.

Then both systems go into a wait state.
The operator determines which system is to assume the role of master via a keylock switch.
The key-operated switch for the update procedure is then actuated, and the master begins updating the standby.

During updating, the user programs in both the master and the standby are not actively processed. There is no output to the I/Os.
If the update procedure terminates successfully, both F systems are automatically linked to the process.
If the update procedure is aborted with error, both systems remain in the passive wait state.
The operator can now either acknowledge the error in both F systems and retry the update, or it can link the master to the process in stand-alone mode.

Passing Control to the Standby When the Master Fails

Before the standby system can take over control of the process, it must have been successfully updated and the two F systems must operate synchronously.

If there is neither a Ready signal nor a master identification signal at the beginning of cycle synchronization or during data interchange, the standby controller assumes that the master has failed and takes over control of the process.

If the master failed during synchronization, any data already transferred are rejected.
The standby controller then takes control of the process using the last valid process input image.

The standby controller now controls the process in stand-alone mode.

Switching the Active Standby Controller to Master Mode

Prerequistes in this case are that the standby controller be in stand-alone mode and actively control the process. The previous master is still not being used, but is ready to go back into operation.

Switching the master switch on the standby controller makes the standby controller the master. After the other F system is put back into operation, it first assumes a passive wait state and signals the master that it is ready to be updated.

The update procedure is started by actuating the key-operated Update switch. During the update procedure, the process is not controlled. Updating must therefore take place under close supervision.

After updating has been completed, the new standby controller must be cyclically synchronized and can then take over should the master fail.

Standby Controller Fails While the Master is Controlling the Process

Prerequisites in this case are that the standby controller has been successfully updated and that the two systems operate synchronously.

If the standby were now to fail, it would simply be disconnected from the process. The master then controls the process alone.

Startup Following Return of Power

If you use a pushbutton to start the process as recommended in Figure 16-3, an automatic restart following a power failure is not possible.

If an automatic restart is desired, you must use a switch in place of the ”STRT” pushbutton and observe the following carefully:

- In order for the master to restart automatically following a power failure, the ”STRT” switch must be in the ”1” position (master will be in stand-alone mode following return of power).
- The ”time-intensive” updating of the standby system can be chosen by the user depending on the process situation.

  This necessitates the following:

  1. Move the ”STRT” switch for process start from ”1” to ”0”.
  2. Initiate the update procedure with the key-operated ”Update” switch.
  3. Move the ”STRT” switch for process start from ”0” to ”1”.

16.2.6 Time Response of FB 230

When speaking of FB 230's time response, a distinction must be made between the required cycle time in cyclic operation and the time needed for the update procedure.

Both times are dependent on the amount of data that must be transferred in the relevant mode.

Time Requirements in Cyclic Mode

FB 230's cycle time requirement in cyclic mode is computed by the approximation method using the following formula:

$$t_z = 60+7.75\times n_z \quad [ms]$$

t_z = Required cycle time in ms
n_z= Number of bytes to be transferred cyclically (input bytes, flag bytes) as per block parameters EF/L, DE/L, MB/L

> **Note**
>
> In the case of failsafe inputs, only half as many bytes are transferred as specified by the EF/L parameter.

Time Requirements for Updating

The time required for updating is computed by the approximation method using the following formula:

$$t_a = 1+0.015\times n_a \quad [sec]$$

t_a = Time for updating in seconds
n_a = Number of bytes to be transferred during updating (data bytes, output bytes, flag bytes) as per block parameters DB1L, DB2L, AF/L, DA/L, DM/L, DBTL

The cycle time required during updating is less than 500 ms
The value specified in DB1 with COM 95F should not be less than 500 ms.

> **Note**
>
> In the case of failsafe outputs, only half as many bytes are transferred as specified by the AF/L parameter.

16.3 Initializing Parameters with COM 95F

In order for FB 230 to function correctly, the following settings must be made with COM 95F:

Configuring the Onboard DIs (DI32/33)

Signal group	=	0
Software interrupt	=	NO
Discrepancy time	=	SHORT 1ms
Short-circuit test, DQ	=	NO

Configuring the Onboard DQs (DQ32)

Signal group	=	0

Configuring the Signal Groups:

Signal group	=	0
Performance on failure	=	STOP

Configuring the Cycle Monitor:

The required setting depends on the user program's cycle time requirements, including the cycle time needed by FB 230 for cyclic data transfers.
The value, however, should be at least 500 ms.

16.4 What To Watch For When Writing User Programs

To make sure that the two systems are operating synchronously, you must observe a number of restrictions and programming techniques when writing your control program.

Always observe the following restrictions carefully:

- No normal timers (timers T1 to T127) may be used.
- Neither the counters (C1 to C127) nor the hardware onboard counters may be used.
- No interrupt inputs may be used.
- No short-circuit-test-outputs may be used.
- No analog output modules may be used.
- No interrupt OBs or time-controlled OBs may be used.
- Only those standard FBs explicitly released for use with the high-availability S5-95F may be used.
- A maximum of two data blocks, each with a maximum of 240 data words, are available for storing program data and program states in the user program.
- Flag bytes beginning with flag byte 200 may be used for temporary purposes only.
- The "passivation of modules" function may not be used in the system.
- No direct I/O access operations (LPY, TPY, LPW, TPW) may be programmed.
- The internal date/time functions may not be used for process control.

Special programming techniques are required for the following functions:

- The use of analog inputs
- The use of SINEC L1 bus interfaces

16.4.1 Structure of a Program for the High-Availability S5-95F

When you write your user program, a particular structure must be used in OB1. OB1 is subdivided into four sections, each of which contains specific program components.
For reasons of clarity, these sections should all end at a network boundary.

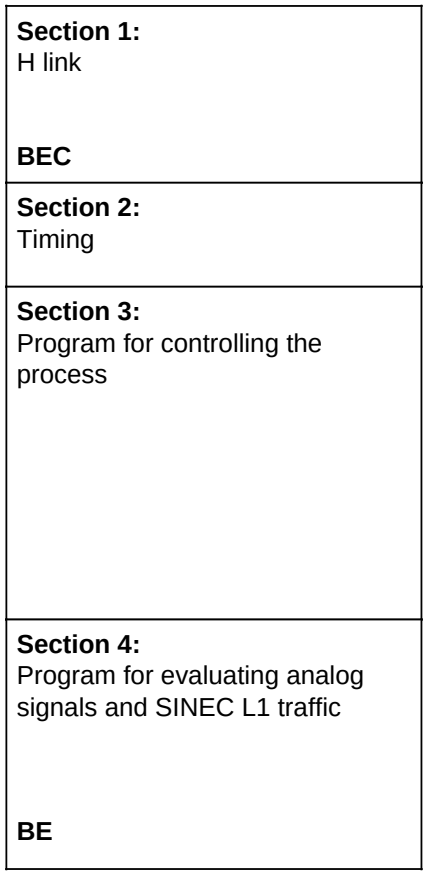


Figure 16-4. Structure of a Program for an H/F System

Section 1:

In this section, that is right at the beginning of OB1, is the FB 230 call for the H link. This section terminates with a BEC statement. The BEC statement has a safety function. Via this statement, FB 230 controls processing of the user program.

Section 2:

The FB 231 call for timing is in this section.

Section 3:

This section contains the control program that controls the process. When programming this section, observe carefully all the restrictions mentioned above. Inputs can be scanned and outputs controlled in this section, but no analog values may be read in nor may SINEC L1 link data be directly evaluated.

Section 4:

All external events which do not reach the system via digital inputs must be handled in this section. This includes, for example, the reading of analog inputs and SINEC L1 link data.

Restrictions:

- In section 4, no outputs may be controlled (DQ, AQ).
- In section 4, parameters used in sections 1 to 3 (I, F, DW, ...) may not be modified.
- Only the flag area that is cyclically exchanged between the two systems (parameter MB/L in FB 230) may serve as interface to area 3.

The following are allowed, but only, of course, under observation of the restrictions mentioned above:

- The use of additional data blocks
- The use of system timers
- Access to analog inputs
- Access to the SINEC L1 Receive buffer
- The use of system counters
- The use of the operating system's date/time functions

Safety Note

The program components programmed in section 4 execute asynchronously in the two systems.
Only the results of the logic operations from these components are synchronized in the two systems.
When programming these components, the user must himself check the overall compatibility with the process.
Only simple and relatively short program sequences are permitted here.

16.5 Timing in the User Program

To ensure synchronicity of the two systems, no normal timer calls may be written in the user program.

For the purpose of synchronization, the interface block transfers a reference time from the master to the standby. To generate the reference time, timer location T0 is used with a time base of 10 ms. The time that elapses between two PLC cycles is shown in both systems in a fixed flag word (REFT parameter in FB 230).

Only this reference time location may be used for timing in the user program. The location must be evaluated in every cycle.

To simplify things for the user, additional function blocks (FB 231, FB 238) are made available which make it possible to simulate an ”extended pulse” timer. Other types of timer can be derived from the ”extended pulse” timer using suitable logic operations.

As many as 64 timers can be simulated with FB 231 anad FB 238.

The digit error in timing in the user program is in the range of the user program cycle time.

Detailed information on the use of blocks FB 231 anad FB 238 can be found in section 16.10.

16.6 Using Analog Inputs in the H/F System

In an H/F system, analog inputs may be scanned only in section 4 of OB1.

An analog signal can be used, for example, for limit monitoring. A binary signal is derived from the exceeding of a limit value. This signal is then allocated to one of the cyclically synchronized flags (see MB/L parameter for FB 230), which is then evaluated in the actual control program.

This ensures that the control programs in the two systems can respond synchronously to the event. The entire analog value can also be allocated to a cyclically synchronized flag word, but many analog values require a large number of cyclically synchronized flags.

To reduce the cycle time requirement, however, the number of cyclically synchronized flags should be kept as low as possible.

The following blocks can be used to evaluate analog signals:

- FB 250 RLG:ANEI Reads a single analog value
- FB 232 RLG:AE2 Discrepancy analysis for two analog values
- FB 233 RLG:AE3 Discrepancy analysis for three analog values

16.7 Using SINEC L1 in the H/F System

In an H/F system, SINEC L1 frame data may be scanned only in section 4 of OB1. Direct access from section 3 of the user program to the L1 Receive DB is not permitted.

If the receive data consist of control commands, they can be directly allocated to cyclically transferred flags (see MB/L parameter for FB 230).
In this case, the frame data must be evaluated and control commands generated from it.

The allocation of the control commands to cyclically transferred flags ensures that the control programs in both systems can respond in sync to the event.

Because the number of cyclically synchronized flags should be kept to a minimum so as not to increase the cycle time any more than necessary, the amount of data which can be transferred via L1 is limited.

Safety Note

- Because of the time response of the SINEC L1 bus system, the same frames may be transmitted twice following failure of the master controller and automatic assumption of control by the standby unit.
- For this reason, SINEC L1 may be used to transfer only those state variables which, when doubled, have no effect on the program function.
  It is, for example, possible to transfer a binary status word whose individual bits are evaluated as control criteria in the program.
- When planning the use of a SINEC L1 bus in conjunction with the H/F system, it must, of course, be carefully observed that both systems have completely symmetrical configurations as well as identical software.
  The SINEC L1 link to a supraordinate system requires two separate bus connections, as both systems must have the same station number.

16.8 Sample Program for Initializing FB 230

Initializing the cyclic data

To stipulate the data to be transferred, a separate start address and block length may be specified in the function block for each of the following data types:
- Failsafe inputs
- Non-failsafe inputs
- Flags to be transferred cyclically

The data to be transferred for each data type must be in one block.

Example:
Failsafe DIs 0.0 to 4.7, non-failsafe DIs 8.0 to 11.7 and flag bytes 4 to 9 are to be transferred in the cyclic program.

Table 16-4. Parameters for Cyclic Data Transfer

Parameter	Value	Description
EF/L	0, 6	0 = Start address 6 = Number of reserved slots
DE/L	8, 4	8 = Start address 4 = Number of reserved slots
MB/L	4, 6	4 = First flag byte 6 = Number of flag bytes to be transferred

Safety Note

All external inputs used in the program must always be transferred cyclically.
Only in this way can the synchronicity between master and standby be ensured.

Initializing the data for the update mode

To stipulate the data to be transferred, a separate start address and block length may be specified in the function block for each of the following data types:

- Failsafe outputs
- Non-failsafe outputs
- Flags to be transferred in update mode
- DB containing the counters
- DB1 containing user data
- DB2 containing user data

For each data type, the data to be transferred must be located in one block.

Example:
Failsafe DQs 12.0 to 14.3, non-failsafe DQs 19.0 to 21.7 and flag bytes 10 to 80 are to be transferred. Updating of the DB containing the counter locations as DB231 with 70 data words, user data block DB1 as DB200 with 120 data words, and user DB2 as DB201 with 240 data words.

Table 16-5. Parameters for Update Mode

Parameter	Value	Description
AF/L	12,4	12 = Start address 4 = Number of reserved slots
DA/L	19,3	19 = Start address 3 = Number of reserved slots
DM/L	10,71	10 = First flag byte 71 = Number of flag bytes to be transferred (80-10+1)
DBTL	231,70	231 = Data block number 70 = Number of data words to be transferred
DB1L	200,120	200= Data block number 120 = Number of data words to be transferred
DB2L	201,240	201 = Data block number 240 = Number of data words to be transferred

Safety Note

In update mode, all external outputs, flags and data used in the control program must be transferred. Only in this way can the synchronicity between master and standby be ensured.

Exceptions:

- Flag area already defined in FB 230's MB/L parameter
- Temporary flags (FY 200 to FY 255)
- Flag word specified in FB 230's FEHL parameter
- Flag word specified in FB 230's REFT parameter
- Output byte reserved by FB 230's LMMA, LDAT, LAKT, STOE, LIVE and LINK parameters

When programming an H/F system, special care must be taken that temporary flags always be preset before using them in a logic operation.

Safety Note

Should a problem occur with the data link, the process output image in the standby controller is reset.
However, only those outputs can be reset which are specified in FB 230's AF/L and DA/L parameters.
Outputs outside these areas are not reset.

Example of a complete OB1 program structure

FB 230 (STL)	Description
OB 1	
NETWORK 1 0000	
0000 :	Section1: H link
0001 :	==========
0002 :JU FB 230	H link
0003 NAME :HF:KOPPL	
0004 EF/L : KY 0,6	Cycl. sync. failsafe DI
0005 DE/L : KY 8,4	Cycl. sync. non-failsafe DI
0006 MB/L : KY 4,6	Cycl. sync. flag bytes
0007 DB1L : KY 200,120	Update user DB 1
0008 DB2L : KY 201,240	Update user DB 2
0009 AF/L : KY 12,4	Update failsafe DQ
000A DA/L : KY 19,3	Update non-failsafe DQ
000B DM/L : KY 10,71	Update flag area
000C DBTL : KY 231,70	Update timer DB for FB 231
000D LMMA : Q 18.0	Master mode indicator
000E LDAT : Q 18.1	Update mode indicator
000F LAKT : Q 18.2	Process control indicator
0010 STOE : Q 18.3	Data link failure indicator
0011 LIVE : Q 18.4	Data trans. failure indicator
0012 LINK : Q 18.5	Inconsistent data indicator
0013 FEHL : FW 2	Int. error bit for diagnostics
0014 QUIT : I 8.0	Input f. failure acknowledgement
0015 STRT : I 8.1	Input f. start stand-alone mode
0016 PUDB : DB 230	Internal buffer Send/Receive
0017 ADIB : KF +1	Updating during operation permitted
0018 REFT : FW 0	Reference timer location for FB 231
0019 TIM1 : KF +40	Timeout value for data link
001A TIM2 : KF +400	Timeout value for data link
001B : BEC	Interlock for user program
001C : ***	

FB230 STL (continued)	Description
NETWORK 2 001D	
001D :	Section 2: Timing
001E :	======================
001F :	
0020 :JU FB 231	Cyclic timer processing
0021 NAME :HF:TBAS	
0022 TDB : DB 231	Timer DB
0023 MAXT : KF +64	Processing of 64 timers
0024 REFT : FW 0	FB 230’s reference timer location
0025 INIT : KF +0	Cyclic operation
0026 :	
0027 :***	
NETWORK 3 0028	
0028 :	Section 3: Control program
0029 :	============================
002A :JU PB 1	User program
002B :JU PB 2	User program
002C :JU PB 3	User program
002D :	...
002E :	...
002F :	...
0030 :	...
0031 :	...
0032 :	...
0033 :	...
0034 :	...
0035 :***	
NETWORK 4 0036	
0036 :	Section 4: Asynchronous Events
0037 :	================================
0038 :	
0039 :JU PB 200	Evaluation L1 link
003A :	
003B :CDB 11	SINEC L1 Receive DB
003C :L DW 1	Control word received
003D :T FW 4	Allocation to cycl. sync. flag
003E :	
003F :	Additional asynchronous events
0040 :	can be scanned and
0041 :	assigned here
0042 :	...
0043 :BE	End of program

16.9 Connecting I/Os to High-Availability S5-95Fs

16.9.1 Connecting Digital Inputs

The following I/O types may be used for digital input modules in an H/F system:

Table 16-6. I/O Types for Digital Input Modules in H/F Systems

I/O Type	Section
A	Sections 4.2.1 and 5.5.1
B	Sections 4.2.1 and 5.5.1
C	Sections 4.2.1 and 5.5.1

Connecting Failsafe Digital Inputs with Single-Channel, Failsafe Sensors

The sensors are wired in parallel on all four subunits. No short-circuit test output is used.

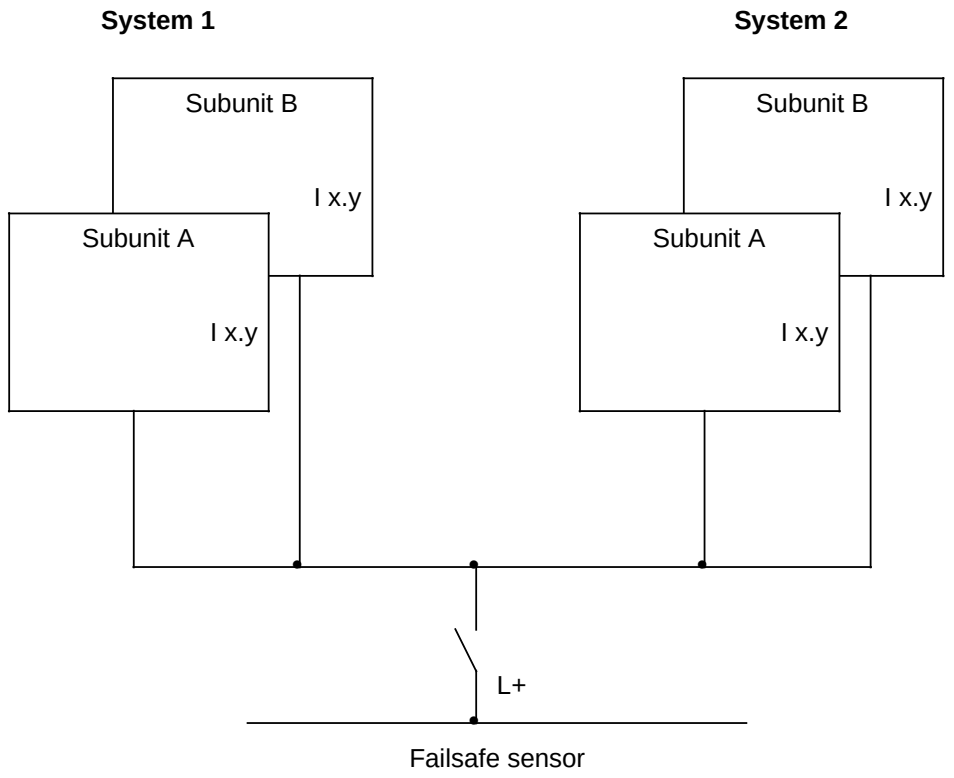


Figure 16-5. Connecting Failsafe Digital Inputs with One Failsafe Sensor

Connecting Failsafe Digital Inputs with Two-Channel, Non-Failsafe Sensors

Sensor 1 is wired in parallel on both systems' subunit A.
Sensor 2 is wired in parallel on both systems' subunit B.
No short-circuit test output is used.

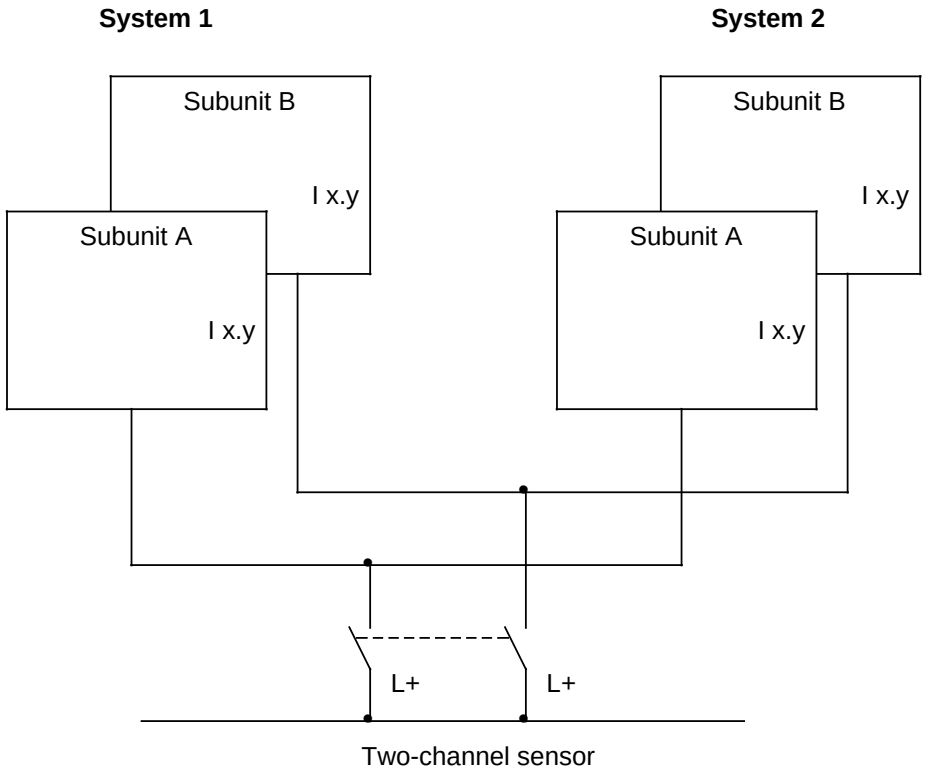


Figure 16-6. Connecting Failsafe Digital Inputs with Two Sensors

Connecting Non-Failsafe Digital Inputs

The sensor is wired in parallel on both systems' subunit A or B.

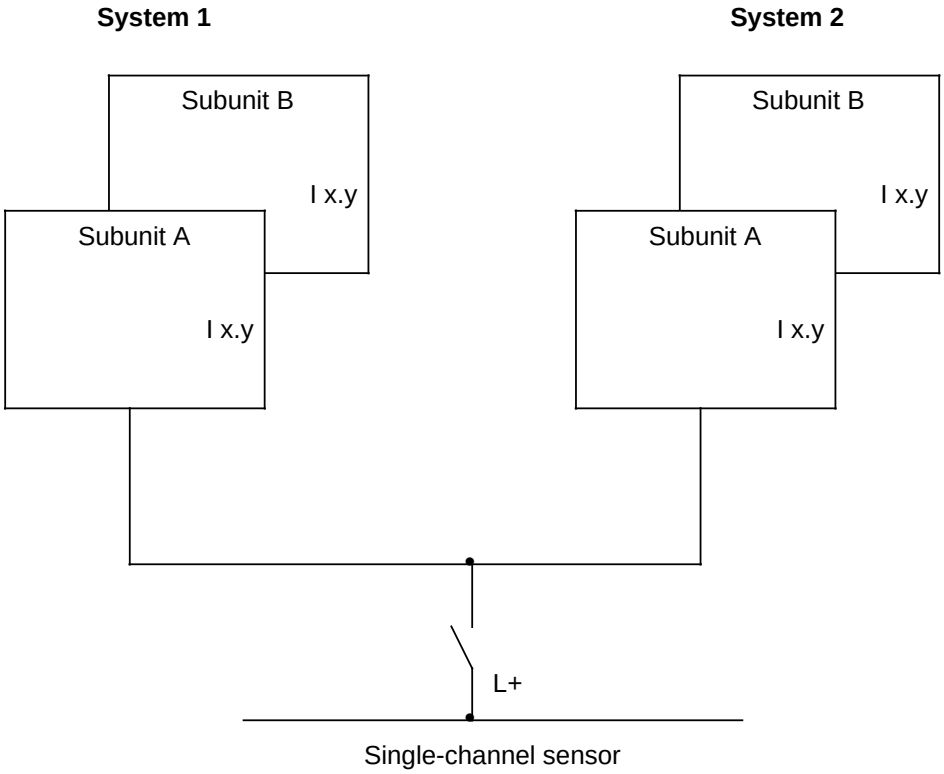


Figure 16-7. Connecting a Non-Failsafe Digital Input

16.9.2 Connecting Digital Outputs

You may use the following I/O types for digital outputs in H/F systems:

Table 16-7. I/O Types for Digital Outputs in H/F Systems

I/O Type	Section
J	Sections 4.2.3 and 5.5.2
K	Sections 4.2.3 and 5.5.2
L	Sections 4.2.3 and 5.5.2
M	Section 5.5.2
N	Section 5.5.2

Connecting Failsafe Digital Outputs with Direct Actuator Control

The M signals and the P signals are connected in parallel on the respective subunits of both systems and decoupled with diodes.

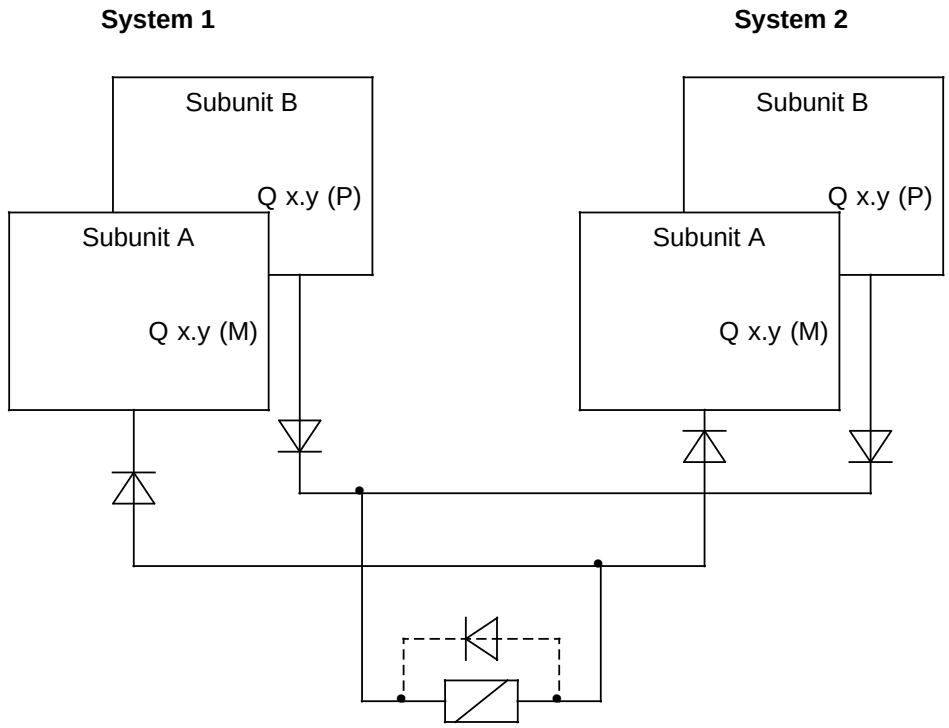


Figure 16-8. Direct Actuator Control in a P/M Circuit

Connecting Failsafe Digital Outputs with Indirect PM Control of the Actuators

Initially, the F controller's outputs control auxiliary relays. Actuator control is handled by NO contacts via a series and a parallel circuit.

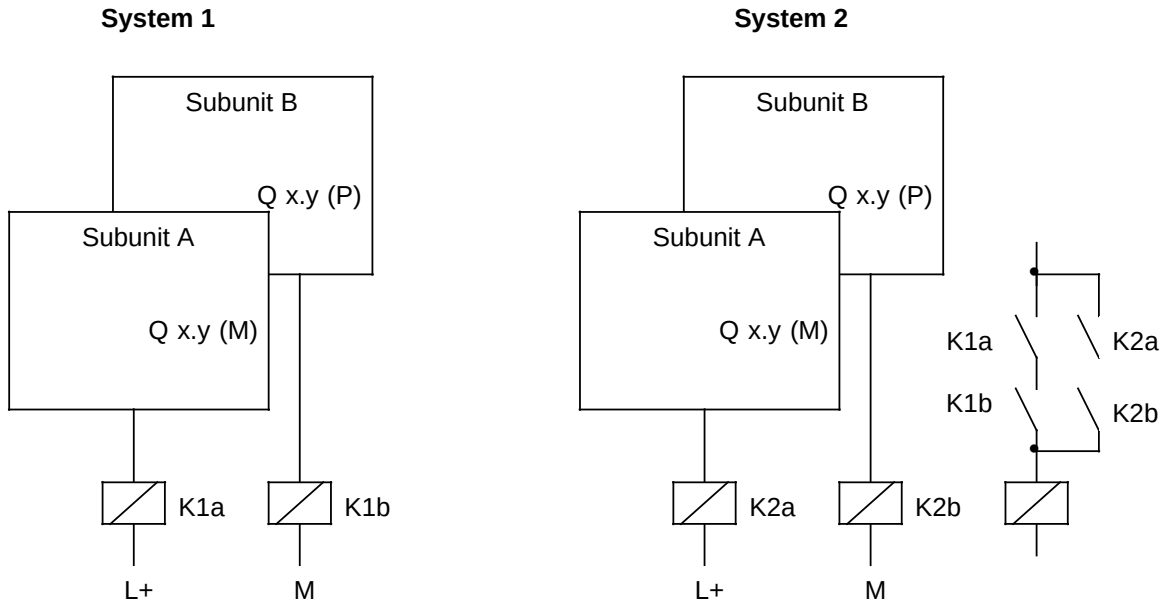


Figure 16-9. Indirect Actuator Control in a P/M Circuit

Connecting Failsafe Digital Outputs with Indirect PP Control of the Actuators

Initially, the F controller's outputs control auxiliary relays. Actuator control is handled by NO contacts via a series and a parallel circuit.

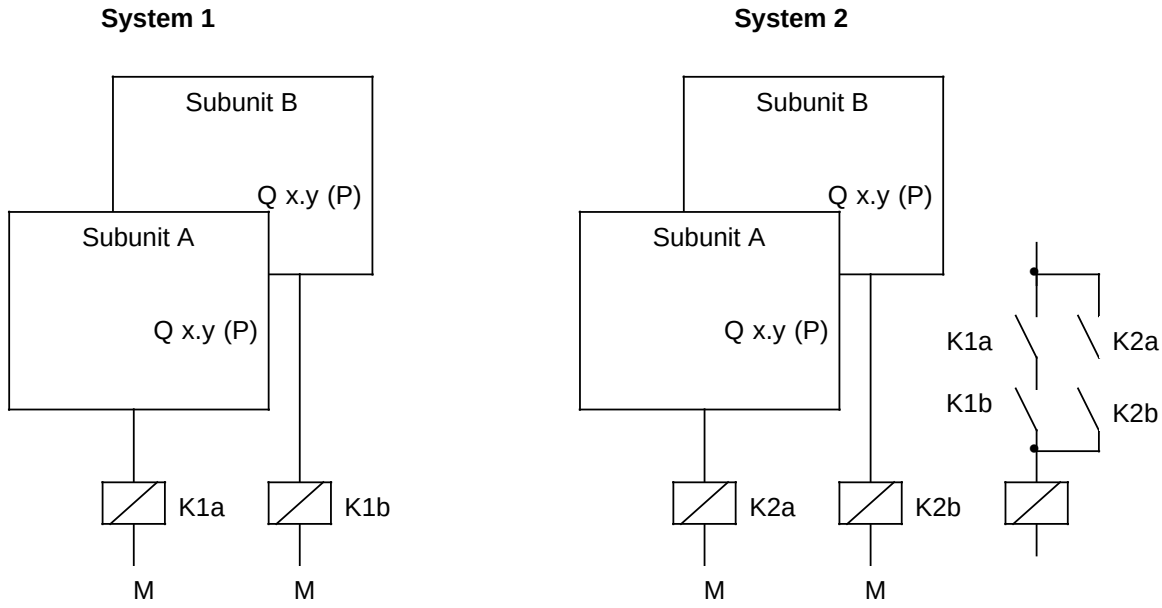


Figure 16-10. Indirect Actuator Control in a P/P Circuit

Connecting Non-Failsafe Digital Outputs with Direct Actuator Control

The L + control of the actuator is wired in parallel on subunit A of both systems or subunit B of both systems and decoupled with diodes.

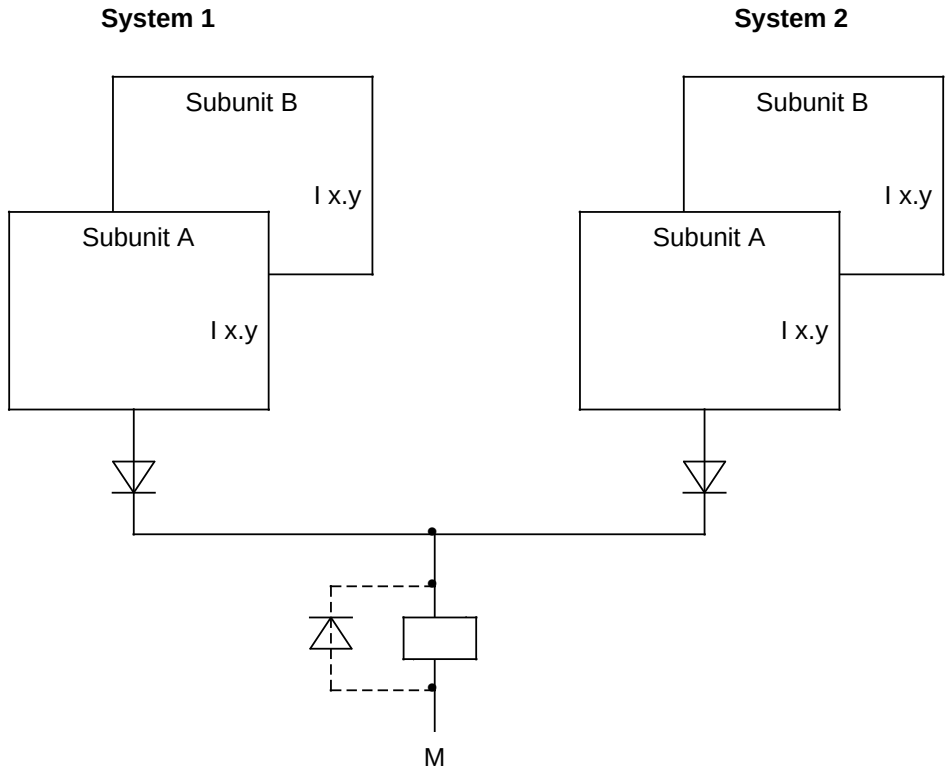


Figure 16-11. Direct Actuator Control for Non-Failsafe DQs

16.9.3 Connecting Analog Inputs

The following I/O types can be used in H/F systems.
Analog signals that are to be read must be connected in series in both systems (current loop).

Table 16-8. I/O Types for Analog Input Modules in H/F Systems

I/O Type	Section
P	Section 11.1
R4.2	Section 11.8.1
R4.4	Section 11.8.2
R6.2	Section 11.8.5
R6.4	Section 11.8.7
R6.5	Section 11.8.8

Connecting Failsafe Analog Inputs with Double Discrepancy Evaluation (I/O Type R4.2)

Example: Sensor 1 is connected to subunit A of both systems.
Sensor 2 is connected to subunit B of both systems.
Current sensors with 4 to 20 mA are used. 6ES5 464-8MG11 modules are used as analog input modules. The analog inputs on subunit A (or subunit B) of both systems are connected in series.

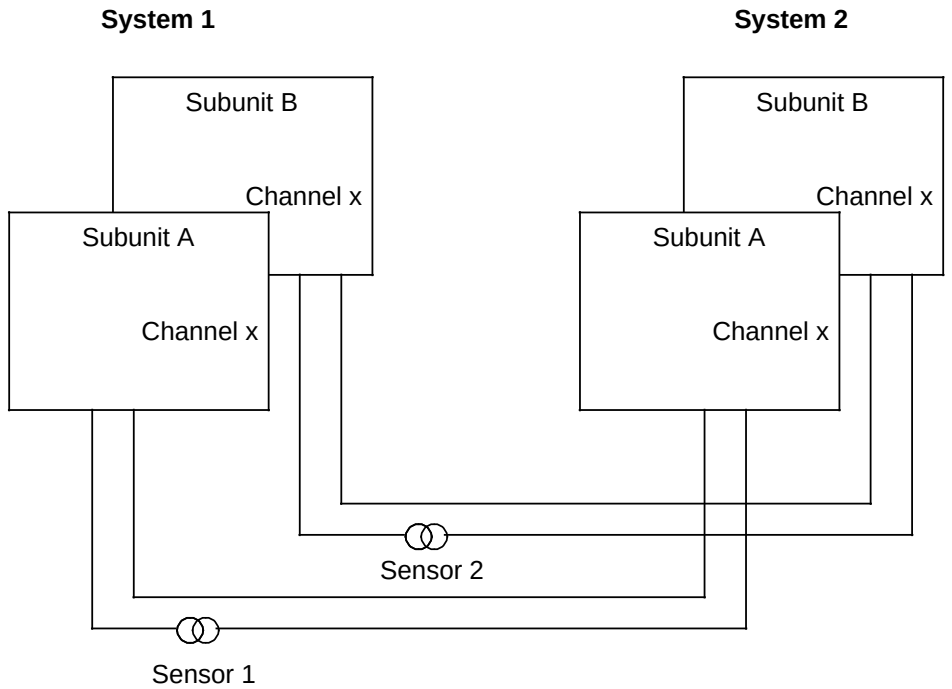


Figure 16-12. Connecting Failsafe Analog Inputs with Double Discrepancy Evaluation

Connecting Non-Failsafe Analog Inputs (I/O Type P)

The sensor is connected to subunit A (or subunit B) of both systems.
A current sensor with 4 to 20 mA is used.
6ES5-8MG11 modules are used as analog input modules. The analog inputs on the subunits are connected in series.

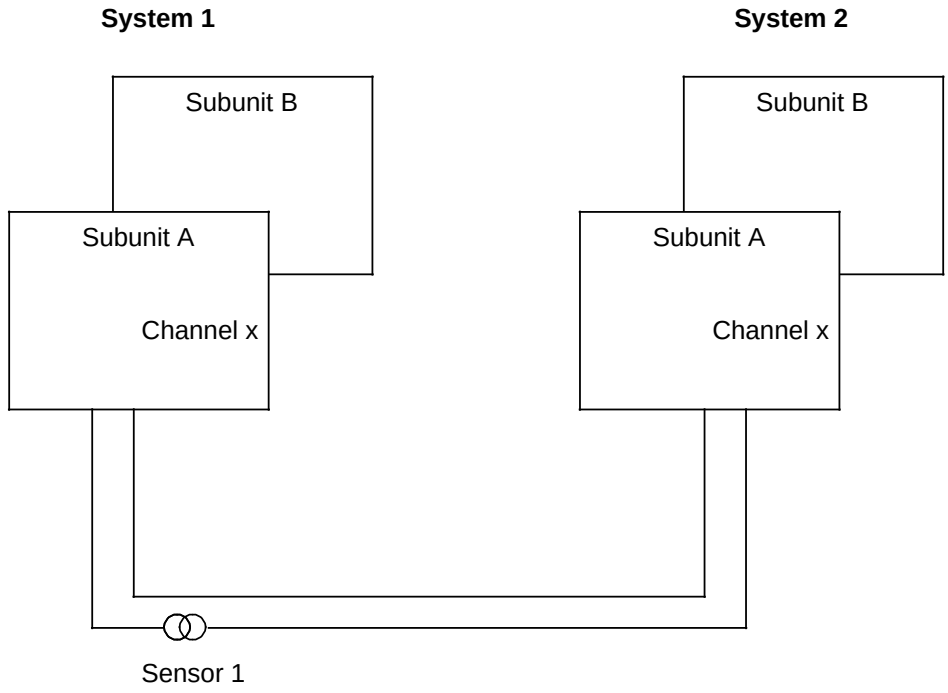


Figure 16-13. Connecting Non-Failsafe Analog Inputs

16.10 Timing for H/F Systems -FB 231 and FB 238-

Function blocks FB 231 and FB 238 are used to generate timers in H/F systems.

When two S5-95F systems are connected with the aid of function block FB 230 (HF:KOPPL) to form a high-availability failsafe system, normal system timers cannot be used in the control program.

FB 231 and FB 238 are provided as replacement. These two function blocks work together, and must always be used together. In addition, a data block comprising 70 data words is also needed to accommodate the timer locations.

FB 231 is used for cyclic processing of the timer locations. It provides a maximum of 64 timers, all of which count synchronously in both F systems.

FB 238 provides a user call for the "extended pulse" timer function.

If other types of timers are required, they must be derived from the extended pulse timer through suitable logic operations.

16.10.1 Block Parameters for FB 231

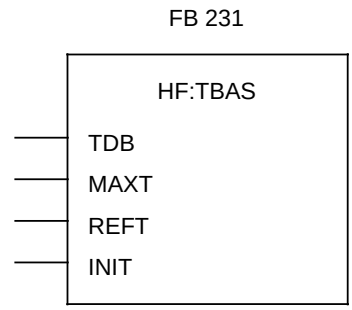
STL	CSF
0000 : 0001 : 0002 :JU FB 231 0003 NAME :HF:TBAS 0004 TDB : 0005 MAXT : 0006 REFT : 0007 INIT : 0008 :BE	

Table 16-9. Block Parameters for FB 231

Parameter	Description	Type		Contents	Remarks
TDB	Data block for timer locations	B		DB 2 to DB 250 Required length 70 DWs	The value assigned to this parameter must be identical to the value assigned to the parameter of the same name in FB 238 (HF:TIMER). Recommendation: DB231
MAXT	Number of timers handled	D	KF	KF = 1 to 64	This parameter can be set to a lower value to save cycle time when fewer than 64 timers are needed in the program. The number of the last available timer is MAXT-1
REFT	Reference timer location provided by FB230	I	W	FW 0 to FW 198	The value assigned to this parameter must be identical to that assigned to the parameter of the same name in FB 230 (H link).
INIT	Initialization mode for calling the block in OB21 (manual cold restart), which resets all timers.	D	KF	KF 0= cyclic operation (OB1) KF 1= initialization (OB21)	FB 231 must be called once with parameter INIT = 1 in OB21 and once cyclically in OB1 with parameter INIT = 0.

Operands Reserved by FB 231

Table 16-10. Operands Reserved by FB 231

Type	Reserved Area
Temporary flags	FY 200 to FY 227

Cycle Time Requirements of FB231

FB231 requires approximately 17.5 ms to process 64 timers at the S5-95F level.

Function of FB 231

FB 231 receives an image of the reference timer location (REFT) from FB 230 from which it forms a data block (TDB) comprising as many as 64 user timer locations. For this reason, the function block must be called precisely once in each OB1 cycle immediately behind the BEC statement after FB 230 (also refer to the description of function block FB 230). The INIT parameter must be 0 at this point.

To initialize the time DB (reset all timers), FB 231 must be invoked once more in the cold restart branch (OB21) with the same parameters, except for the INIT parameter, which must be set to 1.

Safety Note

If the timers are also to be reset after an automatic cold restart following return of power, FB231 must also be called with INIT = 1 in OB22.

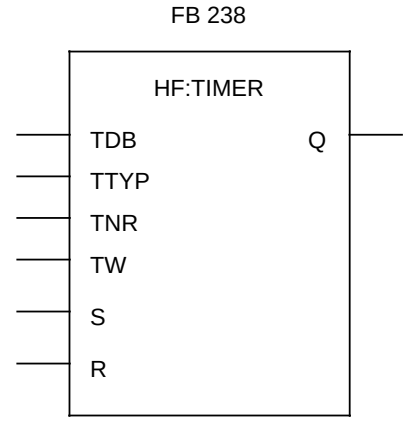
The function block works with a fixed time grid of 100 ms, whereby the maximum digit error is equivalent to the execution time of the user program. The timer locations are updated once per cycle.

Structure of the data block for timer locations:

Table 16-11. Structure of the Data Block for Timers

Data Word	Contents
DW 0	Timer location 0
DW 1	Timer location 1
:	:
DW 63	Timer location 63
DW 64	Last cycle duration in 100 ms units
DW 65	Time increment in 100 ms units
DW 66	Residual carry-over value in 10 ms units
DW 67 to 69	Reserved by FB 238

16.10.2 Block Parameters for FB 238

STL	CSF
0000 : 0001 : 0002 :JU FB 238 0003 NAME :HF:TIMER 0004 TDB : 0005 TTYP : 0006 TNR : 0007 TW : 0008 S : 0009 R : 000A :BE	

Description of the Parameters for FB 238

Table 16-12. Block Parameters for FB 238

Parameter	Description	Type		Contents	Remarks
TDB	Data block for timers	B		DB 2 to DB 250 Required length 70 DWs	The value assigned to this parameter must be identical to the value assigned to the parameter of the same name in FB 231 (Recommendation: DB231)
TTYP	Type of timer function	D	KS	SV=Extended pulse timer	At present, only SV can be specified here; should any other entry be made, none of the timers will be processed.
TNR	Flag byte containing the number of the timer	I	BY	FY 0 to 255	Values from 0 to 63 are valid as timer numbers
TW	Flag word containing the time value in 100 ms units	I	W	FW 0 to 254	Valid time values are numbers from 1 to 32767, which corresponds to 100 ms to 54.61 min
S	Binary input signal for starting the timer	I	BI	I,Q,F	
R	Binary input signal for resetting the timer	I	BI	I,Q,F	
Q	Binary output signal for the timer	Q	BI	Q,F	

Operands Reserved by FB 238

FB 238 reserves no temporary flags, timers or counters. For this reason, temporary flags may be used as transfer parameters, if desired, when the function block is called.

Cycle Time Requirements for FB238

FB238 requires approximately 1 ms for one call.

Function of FB238

The function block provides the "extended pulse" timer function for use in the control program.

The SV function:

The principle of the timer's performance is the of the STEP5 statement SET.

When the signal state at the Start input (S) changes from "0" to "1", the timer (TNR) with the programmed time value (TW) is started. The timer continues to run even when the signal state at the Start input goes from "1" to "0" during the programmed time. The output signal (Q) has signal state "1" as long as the timer is running.

If the signal state of the Start input changes from "0" to "1" while the timer is running, the timer is restarted ("retriggered") with the programmed time value.

When the signal state at the Reset (R) input is "1" while the timer is running, the timer is aborted and the output signal (Q) goes to "0".

Signal time diagram:

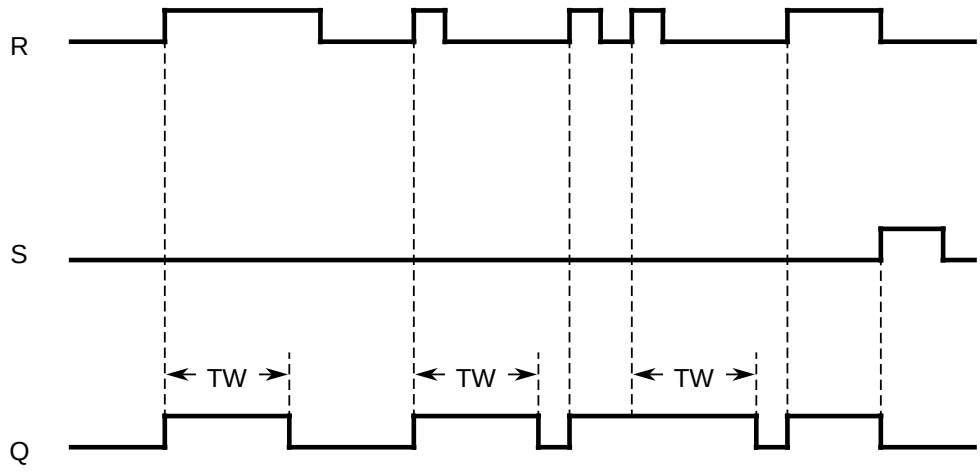


Figure 16-14. Signal Time Diagram

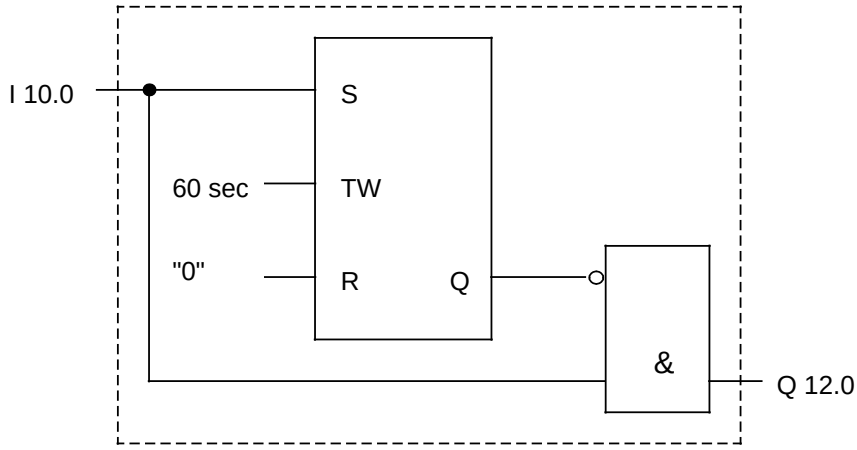
Safety Note

FB 238's functionality requires that FB 231 (HF:TBAS) be called precisely once per OB1 cycle.
The TDB parameter must be identical in FB231 and in all FB238 calls.
Only those timer numbers enabled via the MAXT parameter in FB231 may be used in the control program.

Sample Application for an ON Delay

The signal state of input I 10.0 is to be delayed by 60 seconds before being output to Q 12.0.

Implementation:
The output (Q 12.0) is to go to "1" when the input (I 10.0) is "1" and the delay time has elapsed. The input for resetting the timer is not required here.



Program:

STL	Description
: : :L KB 1 :T FY 100 :L KF +600 :T FW 101 :U F 103.0 :R F 103.0 :JU FB 238 Name :HF:TIMER TDB : DB 231 TTYP : KC SV TNR : FY 100 TW : FW 101 S : I 10.0 R : F 103.0 Q : F 103.1 :A I 10.0 :AN F 103.1 := Q 12.0 : :	 Number of the timer location = 1 Time value = 600 x 100ms = 1 min Reset input for the timer is inactive Timer DB Type of timer function Number of the timer location Time value in 100 ms units Start input for the timer Reset input for the timer Output for the timer Start input Time expired Output of the ON delay

The associated FB 231 calls in OB21 and OB1 may be made as follows.

STL	Description
OB 21/OB 22	
:	
:	
:JU FB 231	
Name :HF:TBAS	
TDB : DB 231	Timer DB
MAXT : KF +64	Number of timers to be processed
REFT : FW 0	FB 230’s reference timer location
INIT : KF +1	Call for initialization
:	
:	
OB 1	
:	
:	
:JU FB 231	
Name :HF:TBAS	
TDB : DB 231	Timer DB
MAXT : KF +64	Number of timers to be processed
REFT : FW 0	FB 230’s reference timer location
INIT : KF +0	Call for cyclic operation
:	
:	

17 Application

17.1	Hardware Prerequisites	17- 1
17.2	Process Description	17- 2
17.3	Installation and Wiring	17- 3
17.4	Entering the User Program	17- 5
17.5	Entering Configuring Data with COM 95F	17- 8
17.6	Testing and Error Simulation	17- 11

Figures

17-1 Process Mimic of a Hydraulic Lifting Device 17- 2
17-2 Setting the Device ID 17- 3
17-3 Circuit Diagram for Simplified Model (with no consideration given to sensor/actuator redundancy) 17- 4
17-4 Sample "Configuring the Onboard DIs (DI 32/33)" Screen Form 17- 8

Tables

17-1 Component Requirements 17- 1
17-2 Timers, Inputs and Outputs 17- 4

17 Application

The following example is that of a model hydraulic lifting device. The purpose of this example is to give you an insight into how the S5-95F works. In order to keep the example simple and easy to understand, we have omitted the many special regulations which would have to be considered when dealing with a real application.

> **Note**
>
> When operating a real application, there are a number of special regulations which must be observed regarding operator input, functional sequence, self-diagnostics and switching elements. A copy of the regulatory standards for a special application can be obtained from the German Statutory Industrial Accident Insurance Institutions (BG) or the German Technical Inspectorate (TÜV).

17.1 Hardware Prerequisites

The following components are required for the sample press:

Table 17-1. Component Requirements

Quantity	Description	Order No.
2	S5-95F basic unit	6ES5 095-8FB01
1	Fiber optic cable, 1 m	6ES5 722-1BB00
2	Backup battery	6ES5 980-0MA11
2	Front connector	6ES5 490-8FB11
1	Standard mounting rail	6ES5 710-8MA11
1	COM 95F software package	6ES5 895-6MF12
3	Momentary-contact switches; NO contacts	
2	Momentary-contact switches; NC contacts	
1	STOP switch	
2	Actuators; lamps (24 V)	

17.2 Process Description

Figure 17-1 shows the process mimic of a simplified hydraulic lifting device. The description of the hydraulic lifting device follows below:

The hydraulic stamp starts to move

- When the stamp is at top dead center
  and
- Momentary-contact start switches S1 and S2 are pressed within 0.5 s. This condition must be fulfilled in order to ensure that the operator is outside the danger area during the process.

The following process executes cyclically as long as S1 and S2 are actuated:

1. The "Lower" command is set and the stamp is lowered.
2. When the stamp reaches momentary-contact limit switch S3, the "Lower" command is reset.
3. The "Raise" command is set and the stamp is raised.
4. When the stamp reaches momentary-contact limit switch S4 (home position), the "Raise" command is reset. The cycle is complete.

Movement is stopped

- When the STOP switch is actuated
- When the operator releases one of the two start switches during the cycle
- When momentary-contact limit switch S4 reaches the "Stamp raised" position

To restart the process, the two start switches must once again be pressed within 0.5 s.

Process Mimic

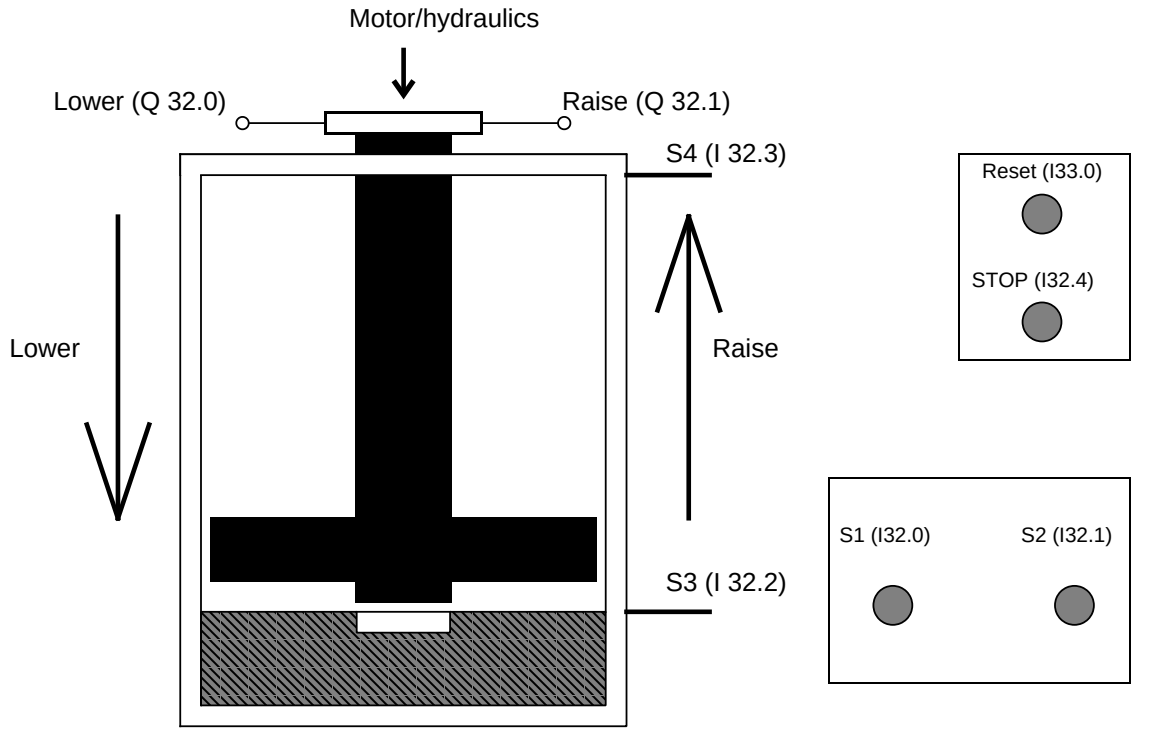


Figure 17-1. Process Mimic of a Hydraulic Lifting Device

17.3 Installation and Wiring

The main prerequisite for error-free S5-95F start-up is proper wiring. To install and wire, proceed as follows:

Affix the two basic units to the standard mounting rail
Insert the 40-pin onboard connectors into the basic units
Remove the protective coverings from the fiber-optics sockets
Connect the two basic units to the fiber-optics cable
Connect basic unit terminals L + and M with +24 V and M
Connect the load voltage supply to the onboard I/O connectors.
To do so, connect
- terminals 1 and 11 with +24 V
- terminals 10 and 20 with M

Insert a backup battery into each basic unit
Set the device ID on both subunits and the length (1 m) of the fiber-optics cable (see Figure 17-2)



Figure 17-2. Setting the Device ID

Connect the sensors as per the circuit diagram.
Care must be taken that the inputs are connected in parallel.
Connect the actuators as per the circuit diagram.
Care must be taken that the output bits on subunits A and B are allocated to different terminals (see section 4.5).
Connect the programmer to subunit A.

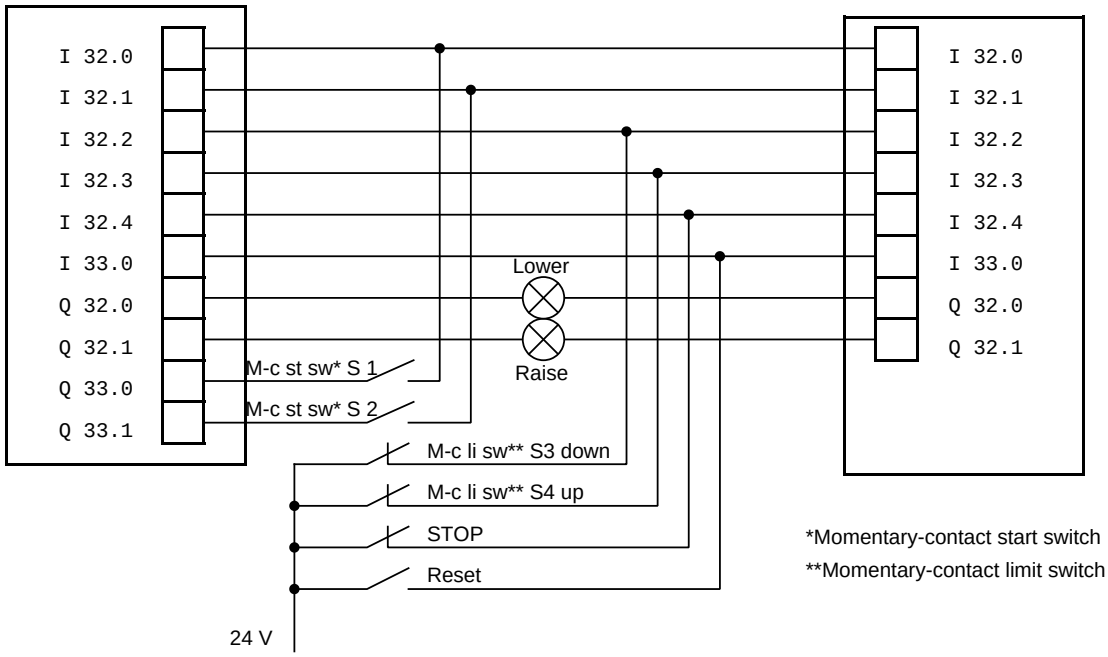


Figure 17-3. Circuit Diagram for Simplified Model (with no consideration given to sensor/actuator redundancy)

Timers, Inputs and Outputs

Table 17-2. Timers, Inputs and Outputs

Operand	Assignment
I 32.0	Momentary-contact switch 1 (S1) for starting the process
I 32.1	Momentary-contact switch 2 (S2) for starting the process
I 32.2	Momentary-contact limit switch (S3) for stamp down
I 32.3	Momentary-contact limit switch (S4) for stamp up
I 32.4	Momentary-contact switch 5 for STOP
I 33.0	Momentary-contact switch 6 for Reset
Q 32.0	Control command for Lower
Q 32.1	Control command for Raise
T1	Timer for momentary-contact switch 1
T2	Timer for momentary-contact switch 2

17.4 Entering the User Program

Before putting the user program onto floppy disk, you must copy the block headers of the integral function blocks to the specified program file.

Switch on both subunits (on/off switch to "I" position).
Result: The subunits are synchronized. The STOP LEDs flicker for about 20 s, then show steady light.
Execute an overall S5-95F reset by making the appropriate entry on the programmer (see section 2.5).
Transfer blocks FB252 and FB255 from the S5-95F to the specified program file.
Enter the user program and store it on floppy disk.

FB252 is invoked in OB1, and executes the short-circuit test, while FB255 passivates an errored byte upon actuation of the RESET button.

OB1 STL	Description
0001 :JU PB 1	Initialization of the parameters
0002 :	or the short-circuit test in FB252
0003 :	
0004 :JU FB 252	Call short-circuit test
0005 NAME :AGF:TEST	
0006 AUFT : FY 110	Job no. 15: Short-circuit test
0007 TYP : FW 112	Irrelevant for job no. 3
0008 NR : FY 115	for Q 33.0 and Q 33.1
0009 BED : F 116.0	Execute test without condition
000A EINZ : F 116.1	Execute test in one cycle
000B PAFE : FY 117	Flag byte for test
000C :	
000D :JU PB 2	Call user program
000E :	
000F :JU FB 255	
0010 NAME :AGF:DIPA	
0011 AUFT : KY 1,5	Depassivate signal group 5
0012 DIPA : I 33.0	with input 33.0
0013 PAFE : FY 104	
0014 :BE	

PB2 CSF	Description

```
SEGMENT 1
           T 1                          Timer T1 for two-handed contemporaneity
            +-----+                     on momentary-contact start switch S2
 I 32.0   --!1_-_ !
 KT 050.0 --!TW DU!-
            !   DI!-
           --!R   Q!-
            +-----+

SEGMENT 2
           T 2                          Timer T2 for two-handed contemporaneity
            +-----+                     on momentary-contact start switch S2
 I 32.1   --!1_-_ !
 KT 050.0 --!TW DU!-
            !   DI!-
           --!R   Q!-
            +-----+

SEGMENT 3
             +---+                      Start condition for Raise (Q 32.1)
 I 32.3   --O!>=1!
 I 32.4   --O!   !   A 32.1
 I 32.0   --O!   !    +-----+
 I 32.1   --O!   !----!R    !
            +---+    !     !
            +---+    !     !
 I 32.2   --0! & !    !     !
 Q 32.0   ---!   !----!S   Q!-
            +---+    +-----+

SEGMENT 4
             +---+                      Start condition for Lower (Q 32.0)
 I 32.4   --O!>=1!                      with retriggering interlock F 1.0
 I 32.2   --O!   !   A 32.0
 I 32.0   --O!   !    +-----+
 I 32.1   --O!   !----!R    !
            +---+    !     !
            +---+    !     !
 T 1      ---! & !    !     !
 T 2      ---!   !    !     !
 I 32.3   --O!   !    !     !  M 1.0
 I 32.4   ---!   !    !     !   +-----+
 F 1.0    --O!   !----!S   Q!---!S    !
            +---+    +-----+   !     !
                      +---+    !     !
          E 32.0   ---! & !    !     !
          E 32.1   ---!   !----!R   Q!-
                      +---+    +-----+ :BE
```

PB1 STL	Description
	Preset values for flags used in FB255
0000 :L KF +15	Job no. 15: Short-circuit test
0002 :T FY 110	
0003 :L KF +0	Relevant for job no. 2 only
0005 :T FW 112	
0006 :L KM 00000000 00000011	Test outputs Q33.0 and Q 33.1
0008 :T FY 115	
0009 :L KF +0	Flag byte for parameter assignment errors and
000B :T FY 116	for test results
000C :BE	

17.5 Entering Configuring Data with COM 95F

Once the blocks have been entered, you must generate data block DB1 with COM 95F. DB1 contains the configuring data for the S5-95F. To generate this block, proceed as follows:

Call COM 95F on the programmer
In the DEFAULTS form, enter the name of the program file containing the user program for the S5-95F
Confirm with <F6>
Enter the configuring target with <F2> "FD configuring"

Configuring the Onboard Inputs for the Short-Circuit Test

Press the following function keys in succession:
- <F2> "IO config"
- <F1> "Onboard"
- <F1> "DI"

Result: The "Configuring the onboard DIs (DI 32/33)" is displayed on the programmer screen:

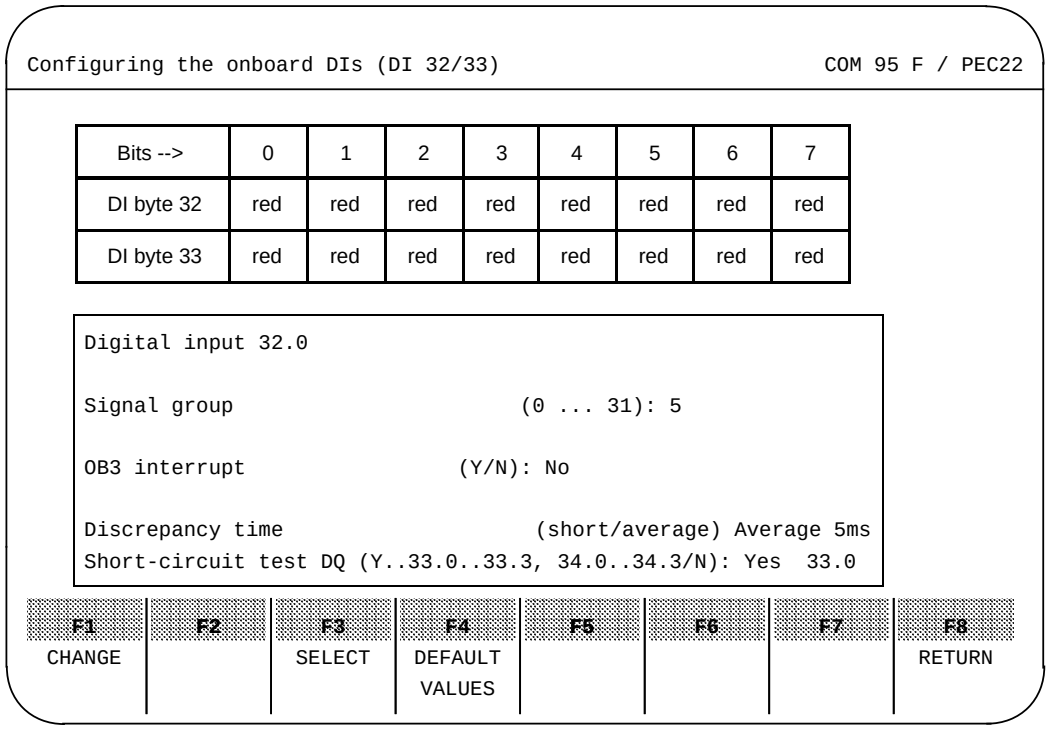


Figure 17-4. Sample "Configuring the Onboard DIs (DI 32/33)" Screen Form

Configuring Inputs I32.0 and I32.1

In the case of a discrepancy on one of the redundant inputs I32.0/I32.1, the S5-95F is to passivate all outputs allocated to signal group 5.
In addition, the S5-95F is to monitor inputs 32.0 and 32.1 for short circuits.

Set the cursor to the field for DI byte 32/column 0 (for bit 0).
Press RETURN.
Result: The cursor is in the input field.
Enter a "5" in the "Signal group" input field.
Choose "Yes" by pressing <F3> in the "Short-circuit test DQ" input field; enter output Q33.0 as test bit.
Press RETURN.
Result: The cursor is in the field for DI byte 32/column 1 (for bit 1).
Press RETURN.
Result: The cursor is in the input field.
Confirm the "5" in the "Signal group" input field.
Choose "Yes" with <F3> in the "Short-circuit test" input field; enter Q33.1 as test bit.
Press ENTER.
Result: COM 95F accepts all entries as made.

Configuring Output Byte 32

Output byte DQ 32 is to be assigned signal group 5.

Select the "Configuring the onboard DQ (DQ 32)" screen form.
Enter a "5" as "Signal group".
Exit the "Configuring the onboard DQ (DQ 32)" form by pressing ENTER.
Result: COM 95F accepts all entries as made.
Terminate configuring of the onboard I/Os by pressing <F8>.
Result: COM 95F accepts all entries as made.

Configuring Signal Groups

If an error occurs in input byte 32 or output byte 32, the S5-95F passivates both bytes.
Signal group 5 is to be configured accordingly.

Select the "Configuring the signal groups" screen form by pressing function key <F3>.
Position the cursor to the field for signal group 5.
Press <F3> to select "P" for "Passivate".
Exit the "Configuring the signal groups" screen form by pressing ENTER.
Result: COM 95F accepts all entries as made.

Terminating Parameter Input

Exit COM 95F by pressing <F8> three times in succession.
Result: COM 95F generates DB1 and transfers it to the specified program file.

Configuring Data for the S5-95F as Generated by COM 95F

DB1 STL	Description
0: KS ='DB1 SDP: SYID 0 DBCON 2 '; 12: KS ='251 CYST N AGCYC 25 DBPA'; 24: KS ='R N ERSI N ERCP N ; TFB:'; 36: KS =' OB13 0 ; CLP: STW N CLK'; 48: KS =' N SET N OHS N OHE N TIS'; 60: KS =' N STP N SAV N CF 0 ; SI'; 72: KS ='G: SGRP Z S ; OBC: CAP 0'; 84: KS =' N CBP 0 N ; OBP: EB 32 '; 96: KS ='0 EB 33 0 EB 59 0 AB 32 '; 108: KS ='0 ; OBHI: HI 59.0 J HI 5'; 120: KS ='9.1 J HI 59.2 J HI 59.3 '; 132: KS ='J ; OBSI: SIN 32.X S N S'; 144: KS ='IN 33.X S N SIN 59.X S N'; 156: KS =' ; OBDE: DE 32.X S DE 33'; 168: KS ='.X S ; KT: DA 33.X E DA '; 180: KS ='34.X E ; PERI: ST 0 S ST'; 192: KS =' 1 S ST 2 S ST 3 S ST 4 '; 204: KS ='S ST 5 S ST 6 S ST 7 S S'; 216: KS ='T 8 S ST 9 S ST 10 S ST '; 228: KS ='11 S ST 12 S ST 13 S ST '; 240: KS ='14 S ST 15 S ST 16 S ST '; 252: KS ='17 S ST 18 S ST 19 S ST '; 264: KS ='20 S ST 21 S ST 22 S ST '; 276: KS ='23 S ST 24 S ST 25 S ST '; 288: KS ='26 S ST 27 S ST 28 S ST '; 300: KS ='29 S ST 30 S ST 31 S ; E'; 312: KS ='XDE: DE Y.X 1L1 ; SL1: P'; 324: KS ='GN N SLN N ; SL1A: KBE N'; 336: KS =' KBS N SF N EF N ; SL1B:'; 348: KS =' KBE N KBS N SF N EF N ;'; 360: KS =' SL1S: UVB N D1S N D1E N'; 372: KS =' D2S N D2E N SNTS1 0 9 S'; 384: KS ='NTS2 0 9 SNTE1 0 9 SNTE2'; 396: KS =' 0 9 TD1S 0 TD2S 0 TD1E '; 408: KS ='0 S TD2E 0 S ; CRC: CRC '; 420: KS ='!??! ; END';	

17.6 Testing and Error Simulation

Prerequisites for the following:

- Both subunits must be switched on and an overall reset executed on each
- The system must be at STOP

Transferring the User Program to the S5-95F

Transfer the user program, including DB1, to subunit A.
The S5-95F's operating system then automatically forwards it to the other subunit over the fiber-optics cable.

Switching on the Controllers

Switch both subunits to RUN.
Result: Red and green LEDs flash briefly. The green LEDs then go to steady light; the S5-95F is in RUN mode.
Should the yellow system event LED go on, use COM 95F to find out what the problem is and carry out the suggested corrective measures.

Error Simulation

You can now test your program as described in the section entitled "Process Description". Should errors occur, you can read out the flags on-line using COM 95F.

Conditions for starting the process:

- The hydraulic stamp is at top dead center
- Momentary-contact start switches S1 and S2 are actuated within 0.5 s (at the same time)
- OFF is not actuated

You can simulate the following errors to test the S5-95F's response:

Simulated Error	Response	Corrective Measures
Short inputs I 32.0 and I 32.1 on one of the subunits and press one of the two start switches.	Yellow error LED goes on. Signal group 5 is passivated.	Eliminate wirebreaks and initiate depassivation by actuating the RESET button.
Pull the fiber-optics connector on subunit B.	Yellow error LED goes on and the STOP LED flashes. S5-95F goes to STOP and tries to synchronize the subunits.	Reinsert fiber-optics connector. Wait until the STOP LED shows steady light. Switch both subunits from STOP to RUN.
Remove the battery from subunit A.	Yellow error LED goes on.	Reinsert the battery and acknowledge error with RESET button.

Note

The Reset button starts FB255, initiating depassivation. Depassivation takes about 30 seconds, after which the S5-95F is fully functional.
Before the S5-95F once again includes the defective component in the cycle, it is tested. This test is particularly apparent when digital outputs are involved, as the LEDs on the digital output modules light up briefly.

18 Rules for Failsafe Operation of an S5-95F

18.1	Acceptance Test for a System Containing an S5-95F	18- 1
18.1.1	Planning Phase	18- 2
18.1.2	Pre-Acceptance Inspection	18- 4
18.1.3	Acceptance Test	18- 5
18.2	I/Os	18- 7
18.2.1	Circuit Diagram for I/Os	18- 7
18.2.2	Discrepancy Times	18- 15
18.3	Operating Modes	18- 17
18.4	Entering the System Identification and ID Number in the System Event DB	18- 17
18.5	Memory Submodules for Safety Mode	18- 18
18.6	Function of the Backup Battery	18- 18
18.7	Retentivity of Timers, Counters and Flags	18- 19
18.8	Requirements for Sensors for Failsafe Digital Input Modules	18- 19
18.9	Requirements for Actuators for Failsafe Digital Output Modules	18- 20
18.10	Response to I/O Errors	18- 21
18.10.1	Passivating of I/Os	18- 22
18.10.2	Depassivating I/Os	18- 23
18.10.3	Standard Value Formation and Reaction at the User Level	18- 25
18.11	Repairs	18- 26
18.12	S5-95F Response Times	18- 27
18.12.1	Response Time and Signal Duration for Cyclical Program Processing	18- 27
18.12.2	Response Time and Minimum Signal Duration for Time-Controlled Program Processing	18- 28
18.12.3	Response Times and Minimum Signal Duration for OB2 Interrupts	18- 29
18.12.4	Response Times and Minimum Signal Duration for OB3 Interrupts	18- 30
18.13	Special Programming Features	18- 31
18.13.1	Disabling/Enabling Interrupts	18- 31
18.13.2	Restrictions on LIR, TIR, TNB and TBS Operations	18- 32
18.13.3	STOP Operation in the User Program	18- 32
18.13.4	Waiting Times in the User Program	18- 33
18.13.5	Scratchpad	18- 33
18.13.6	Post-Loading STEP 5 Blocks in Test Mode	18- 33
18.13.7	Monitoring the Supply Voltage for OB3 Interrupt DI	18- 33

18.13.8 Trigger Edge for OB3 Interrupts 18- 34
18.13.9 Measures for Securing the User Program Against Errors 18- 34

18.14 Addressing and Address Assignments 18- 35
18.14.1 Address Assignments for Onboard I/Os 18- 36
18.14.2 Slot Numbering and Address Assignment for External I/Os 18- 36

18.15 Loadable and Integral Function Blocks 18- 38
18.15.1 Loadable Function Blocks 18- 38
18.15.2 Integral Function Blocks 18- 39

18.16 Programmer-Based Operator Input Functions 18- 41

18.17 Connecting Operator Panels and Text Displays 18- 43

18.18 SINEC L1 LAN 18- 44

18.19 SINEC L1 Safety Times 18- 46
18.19.1 SINEC L1 Safety Time for Receive 18- 46
18.19.2 SINEC L1 Safety Time for Send 18- 49
18.19.3 Response Time During SINEC L1 Traffic 18- 50

18.20 Filter for 24 V DC Power Supply Units 18- 50

18.21 EMC of the IM 316-MA12 . . . . . 18- 51

Figures

18-1 Subdivision of the I/Os into Signal Groups 18- 23
18-2 Schematic of a Structured Program Sequence 18- 24
18-3 Response Time for a Time-Controlled Program Scan 18- 28
18-4 Address Assignment 18- 35
18-5 Consecutive Numbering of Slots in a Single-Tier Configuration 18- 36
18-6 Slot Numbering in a Multi-Tier Configuration 18- 37
18-7 Filter for Safety Regulations to IEC 801-5, Severity Level III 18- 50

Tables

18-1 Circuit Diagram for I/Os 18- 7
18-2 Discrepancy Times for Digital Inputs 18- 15
18-3 Overview of EPROM Submodules 18- 18
18-4 Retentive and Non-Retentive Memory Cells 18- 19
18-5 Blanking Times for Testing Digital Output Modules 18- 20
18-6 Permissible Actuators for the S5-95F 18- 20
18-7 System Response to I/O Errors 18- 21
18-8 Hardware Installation, Removal and Replacement 18- 26
18-9 Impermissible Address Areas for LIR, TIR and TNB Operations 18- 32
18-10 Intervals for AI 464-8MG11 Function Test at the User Level 18- 40
18-11 Programmer Input in Safety, Quasi-Safety and Test Mode 18- 42
18-12 Components for Mains Filter 18- 50

18 Rules for Failsafe Operation of an S5-95F

Safety-related systems are divided into

- Systems requiring no acceptance test
  and
- Systems requiring an acceptance test

The operator is himself responsible for systems not requiring an acceptance test, whereas those requiring an acceptance test are examined by independent experts.

The procedure described in the following is intended as a guideline for the system acceptance test, regardless of whether that test is carried out by an expert or by the operator himself.

The S5-95F is a component of your failsafe system. For this reason, the acceptance test also includes configuring and the S5-95F user program.

Of the utmost importance to the S5-95F acceptance test is the failsafe interplay between all safety-oriented components, such as sensors and actuators.

This section provides

- A guideline for the acceptance test for a failsafe system
- A summary of the topics covered, including added information for the inspection.

18.1 Acceptance Test for a System Containing an S5-95F

Experience has shown that an approval test is best divided into three phases. For this reason, we would suggest dividing the acceptance test into

- A planning phase
- A pre-acceptance inspection
- A system acceptance test

The approval procedure is dependent on the automated process, and is thus individual to each system. In the sections to follow, we would like to provide a guideline to simplify your preparations for the approval procedure.

The COM 95F Software Package Supports the Acceptance and Approval Procedure

The COM 95F software package helps you initialize the system parameters (DB1), and also provides a print function for hardcopy documentation. For more detailed information on the COM 95F software, please see the COM 95F Manual.

> **Note**
>
> It is a requirement of systems which require an acceptance test that the DB1 parameters be initialized with COM 95F. The COM 95F software has a user-friendly menu system, and checks all entries for validity and syntax errors. COM 95F is obligatory to the system approval procedure. The operator/installation engineer must make sure that the COM 95F software package is available to the inspector at the time of the system acceptance test.

18.1.1 Planning Phase

Even while planning your system, you should contact the approving authority and clarify the following points:

Safety Requirements

Find out which standard (e.g. DIN VDE 0116) contains the safety requirements for your system and the class to DIN V 19250.

Risk Analysis

In the risk analysis, the inspector decides which of your system's subprocesses are relevant to failsafety. If some subprocesses are relevant to failsafety while others are not, a risk analysis also becomes necessary for the latter.

Objectives

The inspector decides upon the objectives of your system as regards failsafety, and upon the criteria for a transition of the system to a safe state.
The following questions must be answered:

- In what situations must the entire system be shut down?
- In which cases would it be sufficient to shut down only subprocesses?
- Is the passivation and depassivation of I/Os permitted?

Separation of Failsafe and Non-Failsafe Program Sections:

STEP 5, with its structured block programming, makes it particularly easy to separate failsafe and non-failsafe program sections from one another. The additional convenience of the cross-reference list makes for exceptional user-friendliness, allowing you to easily prove, for instance, that a failsafe input is being processed or that the blocks processed are failsafe blocks, and that failsafe outputs are formed only from failsafe inputs.
In the event of expansions to/retesting of the system, you need only show the inspector the changes made in the individual safe areas. The insusceptibility to changes made in non-failsafe program sections can be easily proven by the cross-reference list.

Decide on Which Timers and Counters are Relevant to Failsafety

The inspector decides which timers and counters are relevant to failsafety. This decision includes values for system parameters which must be initialized with COM 95F, such as

- Maximum discrepancy times
- The OB13 interval

Stipulate system-specific values, such as

- The maximum total response time (response time of the PLC+the response time of sensors and actuators)
- The maximum permissible number of firing attempts (in the case of heating systems)

Hardware Requirements

All modules and subracks which can be used in conjunction with an S5-95F are prototype-tested by the Bavarian Technical Inspectorate, and require no further hardware tests. The modules are subdivided into failsafe modules and reaction-free modules.

Care must be taken that all electrical resources have the necessary properties. Among other things, the inspector tests the following:

- Whether the electrical resources are failsafe or reaction-free
- Whether the technical specifications meet with the required ambient conditions, e.g. temperature and humidity rating

Communication with Other Components

If you plan communications to other components, answer the following questions:

- Who are the partners in the communications link?
- Should data transfer be unidirectional or bidirectional?
- Are safety-related data to be transferred?
- Could data transfer result in falsification of safety-related data in the S5-95F?

Data interchanges with partners in a communication link must in no way impair the failsafety of functions or subprocesses. For this reason, be sure to check all interfaces in a

- Point-to-point connection (e.g. link via a CP 521)
- Bus system (e.g. SINEC L1 or PROFIBUS)

If you want failsafe postprocessing of failsafe data, you always require a failsafe communications link, such as a link via SINEC L1 or PROFIBUS.

The transfer of non-failsafe data requires, at the very least, reaction-free hardware.

Documents for the Pre-Acceptance Inspection

Ask the inspector which documents you are expected to supply at the pre-acceptance inspection.

18.1.2 Pre-Acceptance Inspection

Normally, the documents listed below must be handed over at a pre-acceptance test. The documents must be released by the operator/installation engineer and must include the system, version and date.

- Component mounting diagrams, including specification of the module version
- A printout of the configuring data made with COM 95F, and a data carrier containing these data
- Logic and signal flow diagrams
- Program flow diagrams, program overview and the user program in form of a test file (see the COM 95F Manual), with data blocks on a data carrier and as KOMDOK listing
- A cross-reference list of all inputs, outputs, flags, timers, counters, blocks and I/Os
- Certification report and conditions of the prototype test

Checking of the System Parameters

Use a COM 95F printout to prove that all values relevant to failsafety fulfill the safety requirements.

The monitored cycle time must in all cases be reflected in the required processor response times.

Checking of the Configured Hardware

The inspector uses the component mounting diagram and the printout of the configuring data to make sure that all safety-relevant input and output modules are failsafe. This applies not only to the input and output modules, but also to sensors and actuators.

Checking of the User Program

The inspector checks the user program on the basis of the program listing, the configuring data, the flowcharts and the logic diagrams.

Among other things, he checks

- For proper conversion of the logic diagrams
- The parameter initialization and use of the integral blocks
- The parameter initialization and use of any standard function blocks
- The failsafe formation of the safety-related outputs, timers and counters.
  Outputs, timers and counters are regarded as failsafe when the input parameters used for them are failsafe or when, after viewing an error, it could be proven that the arbitrary fallibility of a non-failsafe input parameter can in no case lead to an unsafe state.

Checklist for the Acceptance Inspection

We would recommend a checklist which includes, among other things,

- Objectives as far as failsafety is concerned
- Function tests for safety-relevant functions
- Conditions of the prototype test
  and
- Conditions of the automated process

18.1.3 Acceptance Test

Before the inspector carries out the acceptance test, you should have tested your failsafe system at least once in failsafe mode with an EPROM submodule.

The following must be made available to the inspector:

- Programmer with STEP 5 basic software package, COM 95F and KOMDOK (KOMDOK is required only when the version of the STEP 5 basic software package is older than 6.0).
- Printer
- At least three EPROM modules of a type approved for failsafe systems
- A UV eraser
- Labels to mark the EPROM submodules

Comparison between Pre-Approved and Implemented Hardware

In order to carry out a hardware comparison, the inspector needs the configuring list, a printout of the configuring data generated with COM 95F, and the component mounting diagrams.

When checking the hardware, the inspector pays particular attention to

- Module conformity (including version)
- The use of failsafe modules for failsafe signals
- I/O wiring
- Links to other devices

Should the comparison show considerable changes, the inspector might well require a new pre-acceptance test.

Comparison between Pre-Approved and Implemented Software

For the software comparison, the inspector requires the current user program, on a data carrier and as KOMDOK listing.

The inspector compares the pre-approved user program with the program actually installed. A comparison with the COM 95F software comparator (see the COM 95F Manual) is recommended.

Should the comparison show considerable differences, the inspector might require another pre-acceptance test.

Error Simulation

Errors should be simulated on the system and on site for all safety-related functions on the basis of the checklist used for the pre-acceptance inspection.

Checking Observance of the Conditions of the Prototype Test

This step is intended to make sure that all conditions of the prototype test and all rules for failsafe operation detailed in the Product Manual have been observed.
This includes, for instance, checking to make sure that the quiescent current principle has been observed for all external safety circuits connected to the system. Other tests include the power supply, the electrical wiring and the memory used in failsafe mode.
Within this framework, a check should also be made to make sure that ambient conditions mentioned in the technical specifictions have been observed.

Documentation

Documentation should include the current user program as listing as well as on floppy disk or EPROM. The EPROM labels should include the following information:

- Installation
- Date
- Signature of the EPROM submodules



Warning

Once the acceptance test has been completed and approval granted, every change to the hardware or software must be discussed with the inspector. Changes made on one's own initiative can result in critical system states.

18.2 I/Os

When connecting I/Os, please observe the rules for I/O addressing discussed in Chapter 6.

Special features regarding the use of I/Os are summarized in the following section.

18.2.1 Circuit Diagram for I/Os

Table 18-1. Circuit Diagram for I/Os

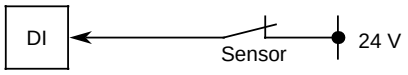
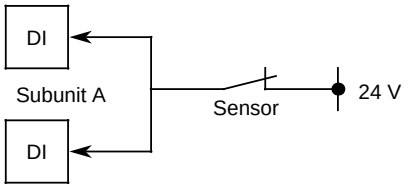
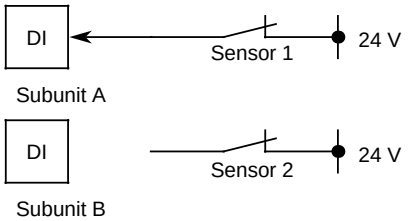
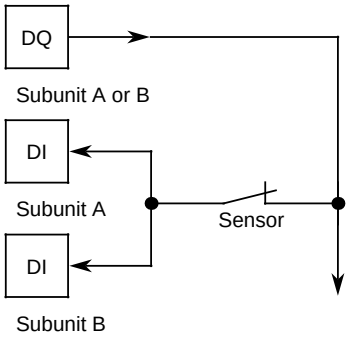
Circuit Diagram	Comments	Modules
I/O type A  Subunit A or B	Non-failsafe binary input module	All standard DI modules from the S5-100U series
I/O type B 	Failsafe binary input module with single-channel failsafe sensor, without line monitoring	2×F DIs 6ES5 431-8FA11 or 2×F DIs Onboard I/Os
I/O type C 	Failsafe binary input module with two-channel sensor, without line monitoring	2×F DIs 6ES5 431-8FA11 or 2×F DIs Onboard I/Os
I/O type D 	Failsafe binary input module with single-channel failsafe sensor and high-quality line monitoring	1×DQs Onboard I/Os 2×F DIs 6ES5 431-8FA11 or 1×DQs Onboard I/Os 2×F DIs Onboard I/Os

Table 18-1. Circuit Diagram for I/Os (continued)

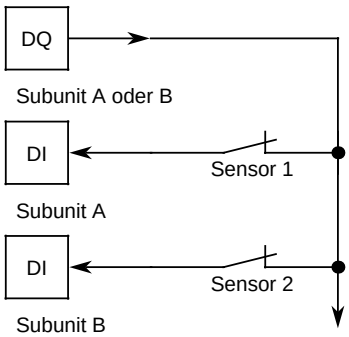
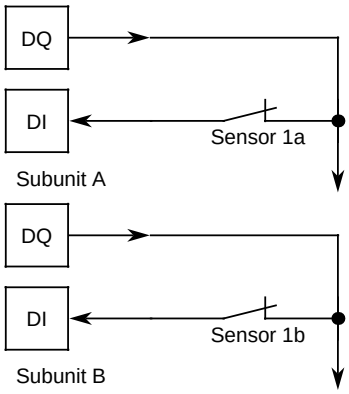
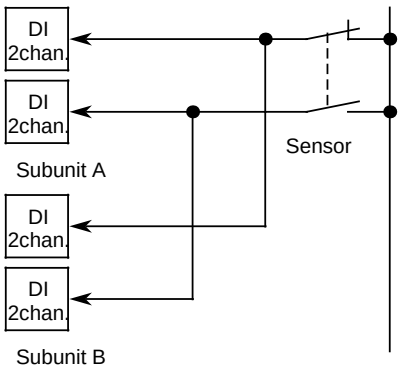
Circuit Diagram	Comments	Modules
I/O type E 	Failsafe binary input module with two-channel sensor and high-quality line monitoring Line monitoring via the same test DQ is possible only when a short between the two signal lines can be totally excluded (e.g. by placing them in separate non-metallic-sheathed cables).	1×DQ Onboard I/O 2×F-DIs 6ES5 431-8FA11 or 1×DQ Onboard I/O 2×F-DIs Onboard I/O
I/O type F (beginning 095-8FB01) 	Failsafe binary input module with two-channel sensor and high-quality line monitoring Line monitoring via two test DQs is required when a short-circuit between the two signal lines cannot be excluded (e.g. when signal lines are placed in a common non-metallic-sheathed cable).	2×DQs Onboard I/O 2×F-DIs 6ES5 431-8FA11 or 2×DQs Onboard I/O 2×F-DIs Onboard I/O
I/O type G.1 	Failsafe binary input module with one non-equivalent sensor and non-equivalence test with function block FB 236 (see section 9.2.1)	4×F-DIs 6ES5 431-8FA11 or 4×F-DIs Onboard I/O

Table 18-1. Circuit Diagram for I/Os (continued)

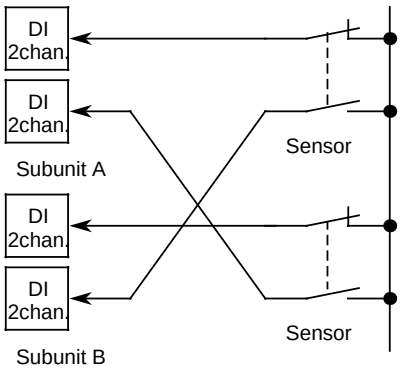
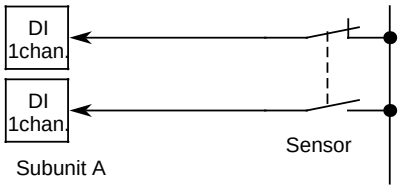
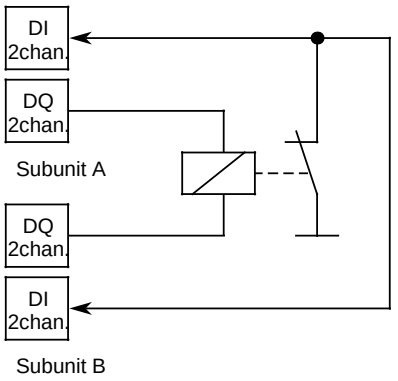
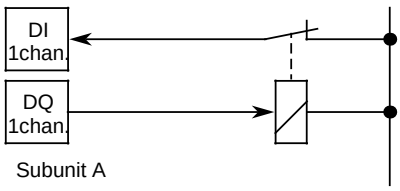
Circuit Diagram	Comments	Modules
I/O type G.2 	Failsafe binary input module with two non-equivalent sensors and non-equivalent test with function block FB 236 (see section 9.2.1)	4×F-DIs 6ES5 431-8FA11 or 4×F-DIs Onboard I/O
I/O type G.3 	Non-failsafe binary input module with one non-equivalent sensor and non-equivalence test via function block FB 236 (see section 9.2.1)	Standard DI module from S5-100U
I/O type G.4 	Failsafe binary output module with equivalence monitoring via function block FB 236 (see section 9.2.1)	2×F-DIs 6ES5 431-8FA11 or 2×F-DIs Onboard I/O
I/O type G.5 	Non-failsafe binary input module with equivalence monitoring via function block FB 236 (see section 9.2.1)	Standard DQ module from S5-100U Standard DI module from S5-100U

Table 18-1. Circuit Diagram for I/Os (continued)

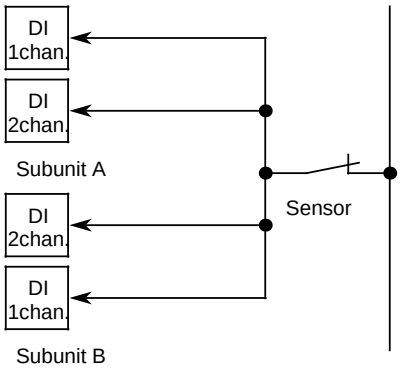
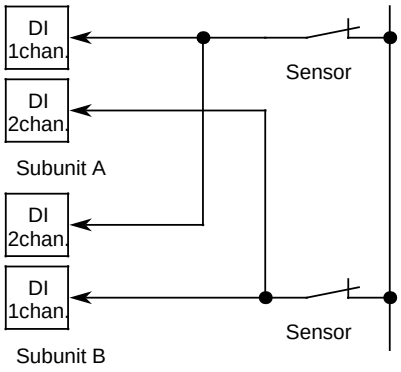
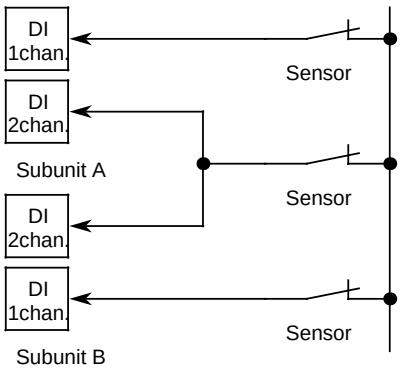
Circuit Diagram	Comments	Modules
I/O type H1 	High-availability, failsafe binary input module with single-channel, failsafe sensor and 2-out-of-3 evaluation via function block FB 234 (see section 9.2.1)	4×F-DIs 6ES5 431-8FA11
I/O type H2 	High-availability, failsafe binary input module with two sensors and 2-out-of-3 evaluation via function block FB 234 (see section 9.2.1)	4×F-DIs 6ES5 431-8FA11
I/O type H3 	High-availability, failsafe binary input module with three sensors and 2-out-of-3 evaluation via function block FB 234 (see section 9.2.1)	4×F-DIs 6ES5 431-8FA11

Table 18-1. Circuit Diagram for I/Os (continued)

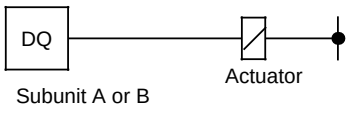
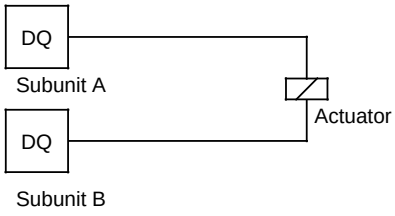
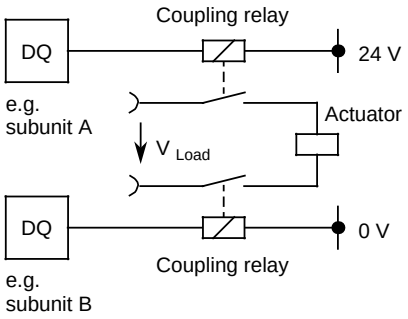
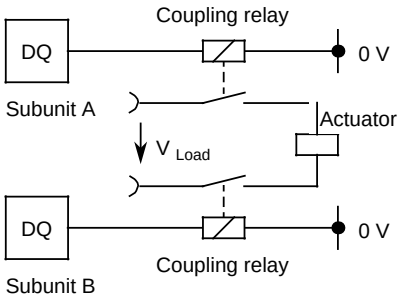
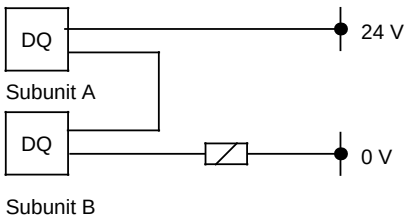
Circuit Diagram	Comments	Modules
I/O type J 	Binary output module, not safety-related	All standard DQ modules from the S5-100U series
I/O type K 	Failsafe binary output module with failsafe actuator	2×F DQs 6ES5 450-8FA11 or 2×F DQs Onboard I/Os
I/O type L 	Failsafe binary output module with indirect driving of a failsafe actuator via contactor/relay contacts (one source output, one sink output) In special cases, coupling relays must be designed for diversity See Table 18-6	2×F DQs Onboard I/Os or 2×F DQs 6ES5 450-8FA11
I/O type M 	Failsafe binary output module with indirect driving of a failsafe actuator via contactor-relay contacts (both modules with source output) In special cases, coupling relays must be designed for diversity See Table 18-6	2×F-DQs 6ES5 450-8FA11
I/O type N 	Failsafe binary output module with failsafe actuator	2×F-DQs 6ES5 450-8FA12

Table 18-1. Circuit Diagram for I/Os (continued)

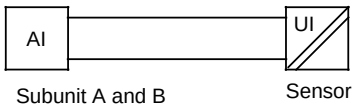
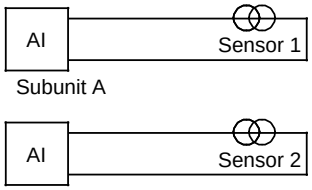
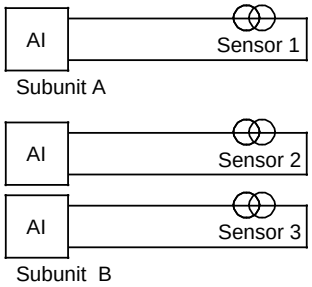
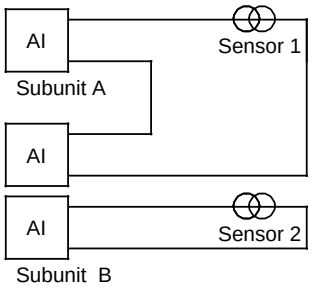
Circuit Diagram	Comments	Modules
I/O type P 	Analog input module, not safety-related	Standard AI modules from the S5-100U series (see Overview in Appendix A)
I/O type R4.2 	Safety-related analog input module (see section 11.8.1)	2×AIs 6ES5 464-8MG11
I/O type R4.4 	Safety-related analog input module (see section 11.8.2)	3×AIs 6ES5 464-8MG11
I/O type R5.1 	Safety-related analog input module (see section 11.8.3)	3×AIs 6ES5 464-8MG11

Table 18-1. Circuit Diagram for I/Os (continued)

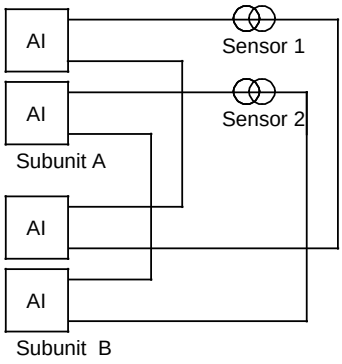
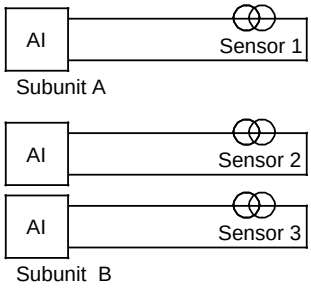
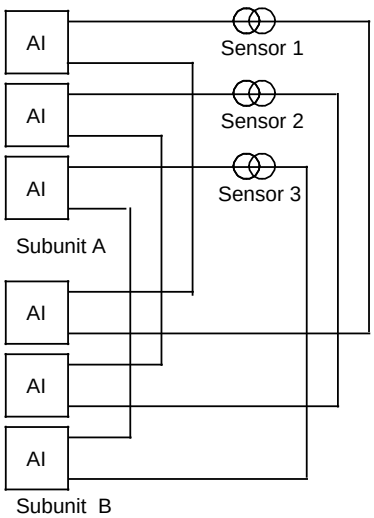
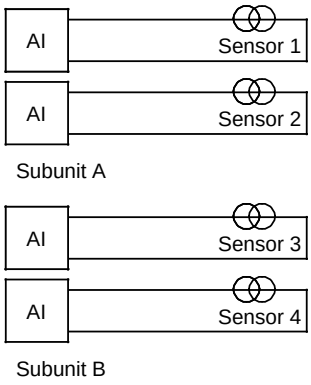
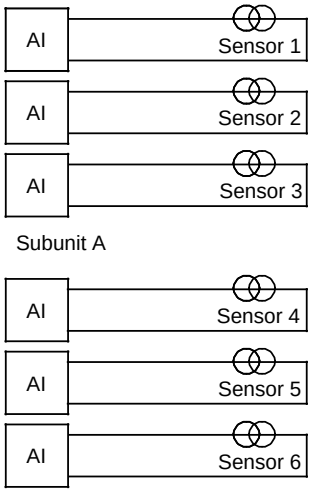
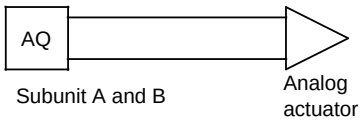
Circuit Diagram	Comments	Modules
I/O type R6.1 	Safety-related analog input module (see section 11.8.4)	4×AIs 6ES5 464-8MG11
I/O type R6.2 	Safety-related analog input module (see section 11.8.5)	3×AIs 6ES5 464-8MG11
I/O type R6.3 	Safety-related analog input module (see section 11.8.6)	6×AIs 6ES5 464-8MG11

Table 18-1. Circuit Diagram for I/Os (continued)

Circuit Diagram	Comments	Modules
I/O type R6.4 	Safety-related analog input module (see section 11.8.7)	4×AIs 6ES5 464-8MG11
I/O type R6.5 	Safety-related analog input module (see section 11.8.8)	6×AIs 6ES5 464-8MG11
I/O type W 	Analog output module, not safety-related	1×AQ 6ES5 470-8MA12 or 1×AQ 6ES5 470-8MB12 or 1×AQ 6ES5 470-8MC12 or 1×AQ 6ES5 470-8MD12

18.2.2 Discrepancy Times

As a rule, the Read result for an input signal is identical in both subunits. However, in certain situations there can sometimes be a discrepancy. Reasons for this may be

- A hardware failure, such as failure of a sensor or input module
- A difference in the instant of access of the two PLCs
- Different switching instants in the case of two-channel sensors or contacts

In order to differentiate between a hardware failure and a fleeting, coincidental signal change, the S5-95F subjects all failsafe input signals to a discrepancy analysis. All you, as user, have to do is provide the permissible discrepancy time in DB1; at the end of this time, at the very latest, the signals must once again coincide. You may define a different discrepancy time for each input byte.

From the instance at which the discrepancy is detected until it has been rectified, or until the specified discrepancy time has elapsed, the two subunits use the last valid value for the user program.

If a discrepancy remains between input signals for a longer period than specified in DB1, the S5-95F assumes a hardware failure and responds with an error message and reacts accordingly.

Safety Note

The discrepancy times must be selected so that the sum of discrepancy time and total PLC response time is less than the error tolerance time of the relevant process function.

Table 18-2. Discrepancy Times for Digital Inputs

Type of Input	Address	Discrepancy Time Programmable in DB1	
OB2 interrupt DI	IB 59	Discrepancy time, short	: Approx.1 ms
OB3 interrupt DI	IB 32 to 33, IB 59	Discrepancy time, short Discrepancy time, average Discrepancy time in	: Approx.1 ms : Approx. 5 ms : n* OB1 cycles
Standard DI	IB 0,2 to 30	Discrepancy time in Discrepancy time in	: n* OB1 cycles : n* OB13 cycles
	IB 32 to 33	Discrepancy time, short Discrepancy time in Discrepancy time in	: Approx.1 ms : n* OB1 cycles : n* OB13 cycles
	IB 59	Discrepancy time, short	: Approx.1 ms

Note

- The average discrepancy time (approx. 5 ms) fluctuates with the synchronization interval. In off-loaded systems, the average discrepancy time is 5 ms, in systems on-loaded to the maximum with interrupts more than 10 ms.
- The disceprancy time in n* OB1 cycles fluctuates by one OB1 cycle.
- The disceprancy time in n* OB13 cycles fluctuates by one OB13 cycle.
- If you select the discrepancy time as the number of OB13 cycles, you must initialize the OB13 interval in DB1. OB13 need not, however, actually be written in the user program.

Power Outage During Discrepancy Analysis

Safety Note

If a power outage/POWER OFF occurs during discrepancy analysis, all the discrepancy timers are reset and started again when the power supply is restored (if the discrepancy is still present).

18.3 Operating Modes

The S5-95F differentiates between three different operating modes:

- Safety mode
- Test mode
- Quasi-safety mode

> **Safety Note**
>
> Whenever the process controller carries out safety functions, the S5-95F must operate in safety mode". The "Test" and "Quasi-safety" modes are for the exclusive purpose of testing the user program.

A detailed description of the operating modes can be found in section 2.5.2.

18.4 Entering the System Identification and ID Number in the System Event DB

Product Identification

Each basic unit has the product identification code "095F$_H$" in DW 0 of the system event DB. Do not change this code.

System ID Number

Whenever you use several S5-95Fs in your installation and operate them with different user programs, there is always the possibility of a mix-up in the EPROM submodules. To avoid mistakes, a unique system ID number can be defined for each S5-95F. This ID number must be entered in DB1 with the COM 95F software when the system parameters are initialized.

In the safety mode, the operating system reads the system ID number from DB1 and transfers it to data word DW 1 of the system event DB. From this point on, the S5-95F accepts only EPROM submodules with this system ID number. You can erase a chosen number by executing a manual overall reset.

> **Safety Note**
>
> When using system ID numbers, always remember that
> - the system ID number may not be zero
> - a manual overall reset sets the system ID number to zero

18.5 Memory Submodules for Safety Mode

The S5-95F operates in safety mode only when the user program is stored on an EPROM submodule. The table below lists the permissible memory submodules.

Table 18-3. Overview of EPROM Submodules

Submodule Type	Submodule Order No.	Capacity	Programming No.
EPROM	6ES5 375-0LA15	8 Kbytes	11
EPROM	6ES5 375-0LA21	16 Kbytes	12
EPROM	6ES5 375-0LA41	32 Kbytes	17
EPROM	6ES5 375-1LA15	8 Kbytes	411
EPROM	6ES5 375-1LA21	16 Kbytes	412
EPROM	6ES5 375-1LA41	32 Kbytes	417

Note

For operation with memory submodules you always requrietwo identical memory submodules. Make sure that both submodules have the same order number.

18.6 Function of the Backup Battery

A backup battery is absolutely necessary for S5-95F operation. The battery is required to maintain the contents of RAM if the power should fail and when the subunits are switched off.

The S5-95F monitors the state of its battery. If the battery charge falls below a certain value during operation, the S5-95F writes a message to the system event DB and subsequently processes OB34 at the beginning of every cycle. If the battery is not replaced within 72 hours, the S5-95F goes to STOP so as not to impair its failsafety.

As soon as the battery no longer has sufficient backup capacity, the yellow battery failure display at the front of the basic unit lights up. When the battery failure display is lit, you can only start the S5-95F by switching the power switch from OFF to ON and then switching from STOP to RUN.
For reasons of safety the S5-95F automatically performs an overall reset and copies the contents of the memory submodule into the internal RAM.

Replacing the Backup Battery

Do not wait until the S5-95F indicates that it is time to change the battery; simply replace it every year as a preventive measure.

Safety Note

Battery replacement while the PLC is at RUN is not allowed in installations which fall under the authority of BIA/BG. In all other installations, replacement of a battery while the PLC is at RUN is permitted only in conjunction with careful observation of the guidelines relating to electrostatic sensitive devices (see Appendix D).

18.7 Retentivity of Timers, Counters and Flags

The table below provides information on the number and retentivity (a timer, counter or flag which is retentive retains its contents, one which is not does not) of timers, counters and flags, data blocks and system data bytes.

Table 18-4. Retentive and Non-Retentive Memory Cells

Operand	Flags, Counters, Timers, Data Blocks, System Data	
	Retentive	Non-retentive
Flags	0.0 to 63.7	64.0 to 255.7
Counters	0 to 7	8 to 127
Timers	_____	0 to 127
Data blocks	1 to 255*	_____
System Data	_____	0 to 255

* Individual data words of DB 252 to 254 are assigned default values by the operating system.

Retentivity of Onboard Counters

The onboard counters are non-retentive, and are reset in the S5-95F's cold restart routine following a power failure.

18.8 Requirements for Sensors for Failsafe Digital Input Modules

The passive error "Sticking of sensor in GO state" must be detectable in the case of analog and static binary sensors relevant to failsafety.

Examples on how to fulfill this requirement:

- Use of prototype-tested sensors under consideration of all additional requirements
- Use of sensors/actuators whose physical properties make such a fault impossible
- Use of redundant sensors/actuators together with organizational measures such as periodic testing. The test cycle must be determined on the process side.
- Automatic error/fault detection (through self-tests or external testing facilities). The efficiency of these tests must be proven.
- Measuring of different process variables for the same purpose (such as pressure and temperature), together with organizational measures such as periodic testing. The test cycle must be determined on the proces side.

The following documents are required for the above-mentioned rating:

- Technical data sheets
  and/or
- Expert opinion or test or certification reports

18.9 Requirements for Actuators for Failsafe Digital Output Modules

In RUN, the S5-95F tests the digital output modules once every hour. To do so, it disables the outputs for a brief period of time, the length of which is different for onboard I/Os than for external I/Os (blanking time).

Table 18-5. Blanking Times for Testing Digital Output Modules

I/Os Tested by the S5-95F	Blanking Time at Output
Onboard digital outputs	< 1 ms
External digital output module DQ 450-8FA11	< 7 ms
External digital output modules DQ 450-8FA12	< 1 ms

With the exception of the (shorter) blanking time, module DQ 450-8FA12 is functionally fully upward compatible to DQ 450-8FA11. For this reason, it is permitted to control an actuator via a module pair consisting of one DQ 450-8FA11 and one DQ 450-8FA12. Note, however, that in this case the controlled actuator must be set for the longer blanking time of the DQ 450-8FA11.

High-speed actuators (such as quick-acting valves) may drop out during the test. If your process cannot tolerate this, choose one of the following three options:

- Use actuators with sufficient lag
- Call the DQ test in DB252 at least once an hour at a non-critical instant
- Wire the actuators to diodes and/or RC networks

Using Actuators with Sufficient Lag

The table below lists a selection of pre-tested coupling relays which do not drop off during the DQ test.

Table 18-6. Permissible Actuators for the S5-95F

Manufacturer	Type	Contacts/ Load Rating
SIEMENS	3TF2010-0BB4	3×9 A at 400 V AC
SIEMENS	3TF4222-0BB4	4×10 A at 230 V AC
SIEMENS	3TF4322-0BB4	4×10 A at 230 V AC
SIEMENS	3TH4262-0BB4	8×10 A at 230 V AC
SIEMENS	3TH4382-0BB4	10×10 A at 230 V AC
SIEMENS	3TH8031-0B	4×10 A at 230 V AC
SIEMENS	3TH3022-0B	4×10 A at 230 V AC
SIEMENS	3TH2022-0BB4	4×4 A at 230 V AC
Télémécanique	LP4-EC09	4×6 A at 230 V AC
ABB	KC22E	4×4 A at 230 V AC
AEG	SH04	4×6 A at 440 V AC

Call DQ Test While System is in a Non-Critical State

FB252 allows you to invoke the DQ test at a defined point in time. Select the instant of the call so that the test will take place while the system is in a non-critical state (for instance because the outputs are "0" or because the brief actuator drop-out has no effect). Note that the DQ test must be invoked in the user program at least once every 50 minutes, as the S5-95F will otherwise invoke it automatically (at a point in time not known to the user).

Connecting Actuators to Diodes and/or RC Networks

By connecting them to diodes or RC networks, you can delay actuator drop-out. The dimensioning of the required components is dependent on the actuator used. If you have any special questions, your contact at the Siemens branch office nearest you would be glad to help.

18.10 Response to I/O Errors

In order to increase its availability, the S5-95F does not always respond to I/O errors by going to STOP. If the inspector is agreeable, you may determine the system response to I/O errors yourself.

Signal group numbers were instated to enable differentiation between I/O errors. These group numbers make it possible to combine process-related I/Os into a signal group.

One of the responses listed in Table 18-7 is assigned to each signal group in DB1.

Table 18-7. System Response to I/O Errors

Programmable Response to I/O Errors	S5-95F Response	Failsafe Response
STOP	System STOP User program is no longer scanned	Failsafe shutdown
Passivation	S5-95F reads all inputs belonging to the signal group as having a "0" signal. S5-95F outputs "0" signal to all outputs belonging to the signal group (outputs are reset).	I/O shutdown
Formation of a standard value via AND operation	S5-95F reads the input signal as "0" and continues the user program scan with this value.	Error report and safety-oriented response in user program.
Formation of a standard value via OR operation	S5-95F reads the input signal as "1" and continues the user program scan with this value.	Error report and safety-oriented response in user program.
Formation of a standard value through reading OLD value	S5-95F reads the input signal in its last valid state and continues the user program scan with this value.	Error report and safety-oriented response in user program.

Initializing the Signal Group for Onboard Counters

> **Note**
>
> When using the onboard counters for safety-related counting tasks, the system response for the selected signal group must be ”STOP” or ”passivation”.
> For non-failsafe functions, you may also choose formation of a standard value after ”AND”, ”OR” or ”OLD VALUE” as system response to onboard counter discrepancies. In the case of a discrepancy in the counters, the S5-95F continues to operate only with subunit A’s counter value.

18.10.1 Passivating of I/Os

If your process consists of several independent subprocesses, you have the option of passivating the I/Os belonging to a signal group.

The passivation of I/Os is understood to mean a software and hardware shutdown of those I/Os. You may passivate I/Os only when they belong to an independent, self-contained subprocess; this must first be carefully examined and ascertained to be the case.

The S5-95F enters an identifier in data words DW35 and DW36 of system event data block DB254 for each pasasivated signal group. These entries can be evaluated in the user program, and the program written to respond accordingly.

The passivation of failsafe I/Os has the following effect:

- DIs (other than interrupt DIs): Immediate removal of the PII entry for the relevant DI
- OB2 interrupt DIs: Immediate resetting of the diagnostic bytes, immediate resetting of DQ byte 32, OB2 will no longer be invoked
- OB3 interrupt DIs: Immediate resetting of the diagnostic bytes, immediate resetting of the PII entry for the relevant interrupt DI bytes
- DQs: Immediate removal of the PIQ entry for the relevant DQ, resetting of the DQ following a direct access operation or end of cycle

> **Note**
>
> Failsafe technology requires that the load voltage for passivated I/Os be interrupted, at the latest, when the secondary error occurrence time has expired. When using an S5-95F, the interruption need not be organized at the user level; instead, the S5-95F acts autonomously, interrupting the voltage via auxiliary shutdown paths (third and fourth paths).

18.10.2 Depassivating I/Os

Faulty sensors and actuators are frequently the cause of I/O passivation. If the problem with sensors and/or actuators can be eliminated while the PLC is at RUN, you can depassivate the I/Os with FB255. The depassivated I/Os are then once again incorporated into the user program.

If possible, you should always depassivate only one signal group. We therefore recommend assigning each signal group a separate depassivation bit. A detailed description on invoking and initializing FB255 was presented in Chapter 9.

> **Note**
>
> Before reactivating the I/Os in a signal group with FB255, you must branch to a routine in your user program in which all variables needed for depassivation are checked, evaluated and, where necessary, updated or the process variables reinitialized.

Note that depassivation with FB255 resets the passivation bit in the system event DB. Actually, this bit is reset even before depassivation has been completed (the I/Os are not yet ready and the parameter assignment error byte (PAFE) shows "Test in progress").

Extending of the PLC Cycle Time and Multiple FB255 Calls

Each time FB255 is invoked, the S5-95F executes only part of the total I/O test. To shorten the test run, it is therefore recommended that FB255 be invoked more than once in the cyclic program. ***FB255 calls*** are allowed only when FB252 is ***not in the process of executing an I/O test*** (DI, DQ, interrupt DI or counter test). You can evaluate FB252's PAFE byte to make sure that this is not the case (PAFE = $D0_H$).

Note that each FB255 call increases the cycle time by approximately 30 ms. The cycle time watchdog timer remains unchanged (i.e. there is no automatic retriggering of the watchdog timer).

Example: Depassivating I/Os

A boiler is heated by four burners. Each burner is independent of the others, so that control of each burner is a separate subprocess. Signal groups 11 to 14 are assigned to the I/Os for these subprocesses.

The control system as a whole is monitored by a preliminary interlock. Since the preliminary interlock applies for all four burners and plays a central role as far as safety is concerned, the I/Os needed for it are assigned signal group 0.

Preliminary Interlock for the Burner Control System I/Os in signal group 0			
Burner 1 I/Os in signal group 11	**Burner 2** I/Os in signal group 12	**Burner 3** I/Os in signal group 13	**Burner 4** I/Os in signal group 14

Figure 18-1. Subdivision of the I/Os into Signal Groups

Diagram for the example

The user program is structured in such a way that when a signal group is passivated, the associated program section will no longer be scanned (see Figure 18-1).
Note that the I/Os in a signal group may not be depassivated until this has been shown to be safe in a check routine at the user level.

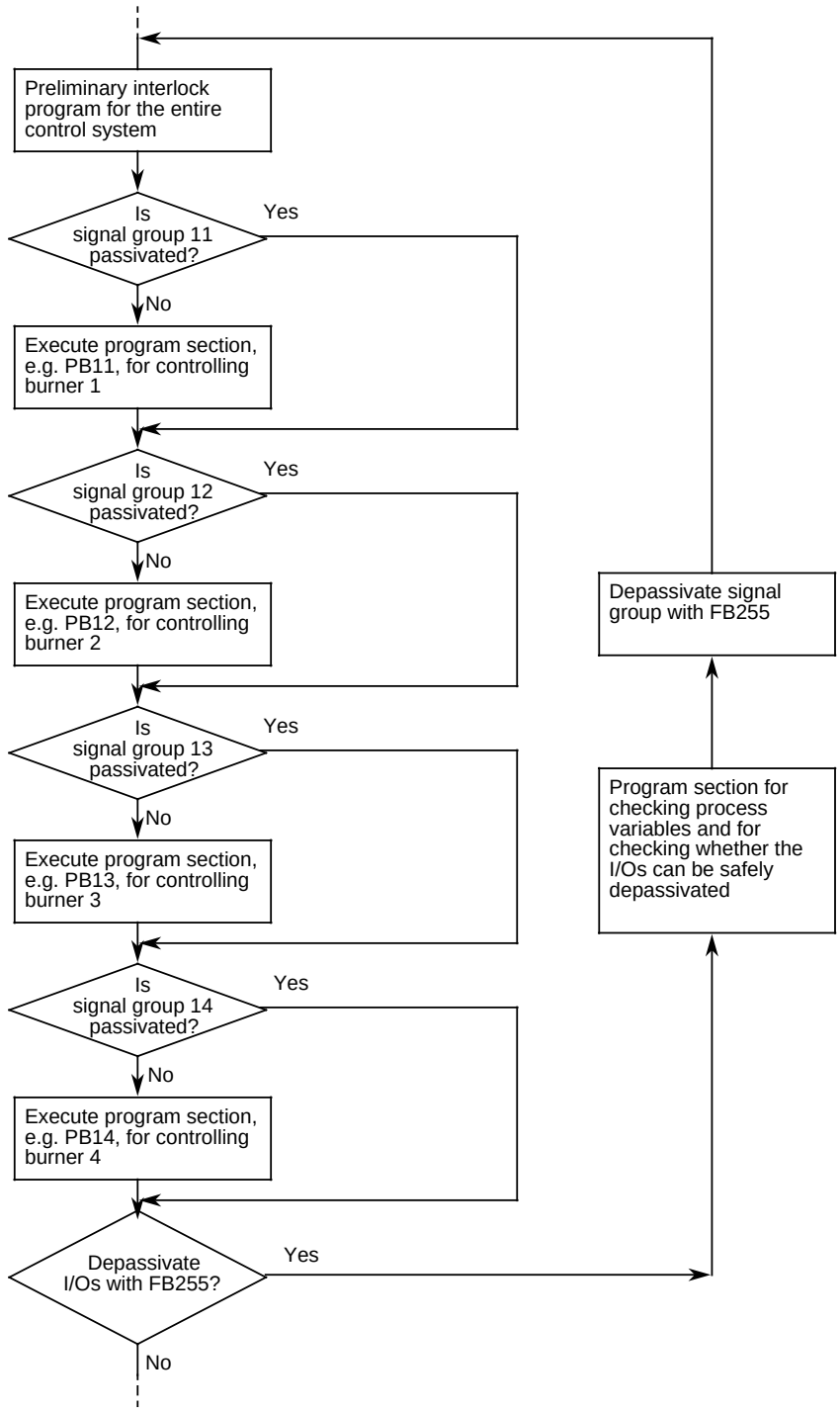


Figure 18-2. Schematic of a Structured Program Sequence

18.10.3 Standard Value Formation and Reaction at the User Level

When processes execute whose immediate shutdown at the first occurrence of an I/O error must be avoided, choose the formation of a standard value as reaction to discrepancies in input signals and initiate that reaction yourself at the user level.

The reaction must be initiated in and monitored by the user program. The program must be written so that it evaluates the error flags in the system event DB and initiates all safety-related responses, such as assuming a safe quiescent state.

There are two possible safety-oriented reactions at the user program level to I/O errors:

When the first I/O error occurs, the process is briefly discontinued, the operating personnel informed (e.g. by an acoustic signal), and an acknowledgement awaited.
When the acknowledgement has been made, the process is resumed in "attended operation" mode and overseen by qualified personnel.
For this purpose, it is necessary to prove that the process can be continued under those conditions by qualified personnel to a switch-off point without any safety risk. The user program must ensure that the process is finally discontinued after expiry of the second error occurence time at the latest.

When the first I/O error occurs, the process is not discontinued. The user program makes sure that the interruptability of the process is maintained, i.e. a second I/O error ***must not*** under any circumstances bring the process to an ***unsafe state***.

Safety Note

If you choose to respond to I/O errors with the ***formation of a standard value and subsequent reaction at the user level***, the responsibility for the failsafe response lies entirely with the operator/installation engineer.

18.11 Repairs

When installing, disassembling or modifying your system, you must proceed as follows:

Table 18-8. Hardware Installation, Removal and Replacement

Installation, Removal and Replacement of:	POWER	Mode	Load Voltage
Basic unit	Supply voltage OFF	Irrelevant	Irrelevant
Bus units I/O interface modules 40-pin onboard I/O connectors 9-pin sub D connectors for interrupt and counter inputs	POWER OFF	Irrelevant	Irrelevant
I/O modules	Irrelevant	STOP/RUN*	Irrelevant

* When I/O modules are inserted or removed in RUN mode, the S5-95F disables the outputs for approximately 300 ms. Moreover, voltage dips may ensue, causing the S5-95F to go to STOP.

Safety Note

If you modify the bus configuration while the S5-95F is at RUN, the PLC initializes the bus. This takes approximately 300 ms, and causes disabling of the outputs for the same length of time. Under worst-case conditions, it is also possible that

- there will be voltage dips on the internal bus which may cause the S5-95F to go to STOP
- execution of OB13, if pending at the time, will not take place punctually, causing the S5-95F to go to STOP

Safety Note

An overall reset must not be performed until all faulty components have been repaired, and is the sole responsibility of the user!
In failsafe systems, an overall reset is considered relevant to a system's failsafety because the error info in the system event DB is lost when an overall reset is executed.

Safety Note

When I/O modules are inserted and removed, the contacts of bus units 700-8MA11 and 700-8MA21 are exposed and unprotected. Remember that, in this case, the S5-95F's EMC is reduced, and the strict requirements to DIN IEC 801-2 are no longer fulfilled.
When handling modules, always be sure to follow the guidelines and regulations for the prevention of electrostatic charge with the utmost care (see Appendix D).

18.12 S5-95F Response Times

The maximum response times must always be computed for systems and processes which require approval. The maximum response times must be less than the maximum values stipulated by the rules and conventions and by the qualified experts.

The following subsections discuss the S5-95F's response times under worst-case conditions. In addition to the response times described here, factors such as the delay times for sensors and actuators and the delay times of mechanical parts such as the stamp in our example.

18.12.1 Response Time and Signal Duration for Cyclical Program Processing

If a change in an input signal results in a change in an output signal, the S5-95F's response time is defined as the moment at which the signal changes at the S5-95F's input terminals until the moment at which the signal changes at the S5-95F's output terminals.

In quasi-safety mode, the S5-95F responds with STOP when the maximum cycle time specified in DB1 is exceeded. As a result, the maximum response time $T_{OB1\ response}$ is computed as follows:

$T_{OB1\ response}$=2×max. PLC cycle time 1)

1) You must specify the value for the max. PLC cycle time in DB1

The use of the programmer's STATUS function in Test mode increases the response time. In the worst case, a response time $T_{OB1\ response}$ of up to 10 times the maximum PLC cycle time is a possibility.

As a rule, the response times to process changes are lower. Please refer to section 7.4.3 for more information.

Increased Response Time Based on the Programmer's STATUS Function

Note

When it executes the STATUS function, the S5-95F processes the specified block in a special mode; the block's runtime is thereby considerably increased, and is operation-dependent. Please note that this also results in an increase in both the PLC cycle time and the response time. For this reason, keep your blocks as short as possible.
The formula shown above remains the basis for the maximum response time for $T_{OB1\ response}$.

Minimum Signal Duration During Execution of OB1

Note

During a cyclic program scan, the S5-95F detects a signal change under worst-case conditions only when signal levels "0" and "1" remain present for ***at least one OB1 cycle***.

18.12.2 Response Time and Minimum Signal Duration for Time-Controlled Program Processing

If a change in an input signal results in a change in an output signal, the S5-95F's response time is defined as the instant at which the signal changes at the S5-95F's input terminals until the instant at which the signal changes at the S5-95F's output terminals.

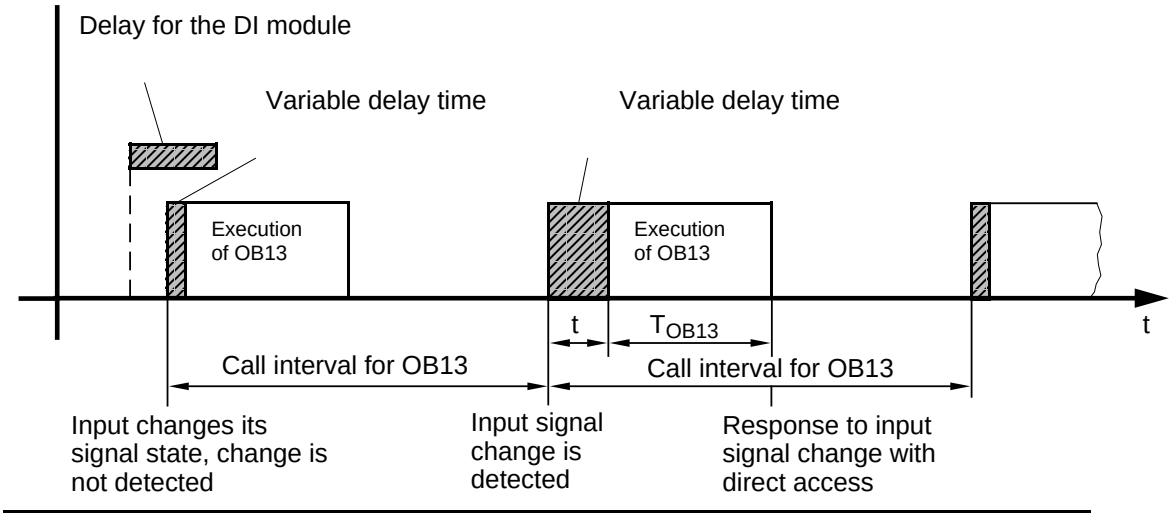


Figure 18-3. Response Time for a Time-Controlled Program Scan

The maximum response time for a time-controlled program scan depends on:

- the specified OB13 call interval
- delay time t (see section 7.4.3)
- OB13's scan time T_{OB13}
- the input module's delay time
- the data cycle (if external I/O modules are used)

Under worst-case conditions, the following applies for the response time in a time-controlled program scan:

$T_{OB13\ response}$ 1×OB13 call interval
+ OB13 scan delay (max. one OB13 call interval)
+ OB13 execution time, plus 1 data cycle when the S5-95F is configured with external I/Os and 1 data cycle when a direct access operation to external I/Os is programmed in OB13 (max. one OB13 call interval)
+ Delay time for the input module

For additional information, please refer to sections 7.4.4 and 7.4.5.

Minimum Signal Duration During Execution of OB13

> **Note**
>
> In a time-controlled program scan (OB13), the S5-95F detects a signal change under worst-case conditions only when signal levels "0" and "1" are present for ***at least one OB13 call interval***.

18.12.3 Response Times and Minimum Signal Duration for OB2 Interrupts

The interrupt response time is the time between an edge change at the terminals of an interrupt input and the signal change at the terminals of an output on the base unit. Not included in the interrupt response time are the delays caused by circuits, sensors and actuators.

Interrupt Response Time for OB2 Interrupts

The interrupt response time for OB2 interrupts $T_{OB2\ response}$ depends on

- the S5-95F's internal delay for OB2 interrupts (max. 5.6 ms)
- execution time T_{OB2} of organization block OB2 (should not exceed 2 ms; see section 12.2.1)

In the majority of applications, the programs in OB2 are relatively short and the execution time of OB2 is less than 0.2 ms. You can expect a ***typical interrupt response time*** of 3 ms for OB2 interrupts.

Typical Interrupt Response Time for OB2 Interrupts

If you have not disabled an interrupt (IA/RA operations) in the user program, the ***typical interrupt response time*** is calculated as follows:

$T_{OB2\ response,\ typical}$ 2.8 ms+T_{OB2}

However, if you disable an interrupt in the user program, the interrupt response time will be extended by the execution time of the program sequence for which the interrupt was disabled (max. 5 ms).

Interrupt Response Time for OB2 Interrupts under Worst-Case Conditions

If you have not disabled an interrupt (IA/RA operations) in the user program, the ***interrupt response time under worst-case conditions*** is calculated as follows:

$T_{OB2\ response,\ worst\text{-}case}$ 5.6 ms+T_{OB2}

However, if you disable an interrupt in the user program, the interrupt response time will be extended by the execution time of the program sequence for which the interrupt was disabled (max. 5 ms).

Minimum Signal Duration for OB2 Interrupts

Note

An OB2 interrupt is triggered by a negative edge at an OB2 interrupt input. The S5-95F detects the edge under worst-case conditions only when signal levels "0" and "1" have a duration of ***at least 4 ms***. For this reason, be sure to use suitable sensors (no fleeting contacts or chattering switches).

18.12.4 Response Times and Minimum Signal Duration for OB3 Interrupts

The interrupt response time is the time between an edge change at the terminals of an interrupt input and the signal change at the terminals of an output on the basic unit. Not included in the interrupt response time are the delays caused by circuits, sensors and actuators.

Interrupt Response Time for OB3 Interrupts

The interrupt response time for OB3 interrupts $T_{OB3\ response}$ depends on:

- The S5-95F's internal delay for OB3 interrupts (max. 12 ms if you do not process OB2 interrupts and max. 16 ms if you process both OB2 and OB3 interrupts)
- Execution time T_{OB2} for organization block OB2 (should not exceed 2 ms; see section 12.2.1)
- Execution time T_{OB3} for organization block OB3 (should not exceed 8 ms; see section 12.3.1)
- Number of interrupt input bytes n_{bytes} used for processing OB3 interrupts

In the majority of applications, the programs in OB2 and OB3 are relatively short. The execution time of OB2 is generally less than 0.2 ms and that of OB3 less than 1 ms. You can therefore expect a ***typical interrupt response time*** of 8 to 14 ms for OB3 interrupts.

In individual cases, the interrupt response time for OB3 interrupts may be prolonged:

OB3 Interrupts Only, No OB2 Interrupts: Interrupt Response Time for OB3 Interrupts in Individual Cases

Under worst-case conditions, the response time for OB3 interrupts if no OB2 interrupts are used is

$T_{OB3\ response,\ wort\text{-}case}$ $12\ ms+1.25\times T_{OB3}+1.5\ ms\times n_{bytes}$ (n_{bytes}=1, 2 or 3)

Both OB2 and OB3 Interrupts: Interrupt Response Time for OB3 Interrupts in Individual Cases

Under worst-case conditions, the response time for OB3 interrupts when OB2 interrupts are also used is

$T_{OB3\ response,\ worst\text{-}case}$ $16\ ms+5\times T_{OB2}+1.25\times T_{OB3}+1.5\ ms\times n_{bytes}$ (n_{bytes}=1, 2 or 3)

Delay of OB3 Processing During a Short-Circuit Test

During a short-circuit test, execution of OB3 can be delayed by up to 10 ms (S5-95F disables the interrupts during the test).

Minimum Signal Duration for OB3 Interrupts

> **Note**
>
> An OB3 interrupt is triggered by an edge at an OB3 interrupt input or when the comparison value of an onboard counter is reached.
> The S5-95F detects an edge at an OB3 interrupt input under worst-case conditions when the signal states "0" and "1" have a duration of ***at least (5 ms + $T_{OB3\ response}$)***; in the case of counters, signal levels "0" and "1" must have a ***minimum duration of 0.5 ms***. For these reasons, be sure to use suitable sensors and keep the OB3 runtime to a minimum.

18.13 Special Programming Features

To ensure the failsafety of systems which require approval, a number of STEP 5 operations have been restricted or may not be used at all.

18.13.1 Disabling/Enabling Interrupts

Note that the STATUS function cannot be used on blocks which are invoked between an IA and an RA operation in the quasi-safety mode.

> **Safety Note**
>
> Note that the interrupt response time may increase by the duration of the interrupt disable. If possible, never disable interrupts for more than 5 ms. If they are disabled for a longer period, the S5-95F responds as follows:
> - After 5 ms, the S5-95F reports error 13, i.e. "Error in user program, interrupts disabled too long". There is no other response (the system remains at RUN).
> - After 10 ms, the S5-95F reports error 4, i.e. "Too many interrupts", and goes to STOP to protect its failsafety.
>
> If your process requires a system response after 5 ms, you must program it yourself. Single out the error with the number 13 by evaluating the system event DB in OB37.

18.13.2 Restrictions for LIR, TIR, TNB and TBS Operations

The S5-95F monitors attempts to access address areas to which the user has no access.

Table 18-9. Impermissible Address Areas for LIR, TIR and TNB Operations

Operation	Restriction
LIR	Access is forbidden to address areas $4F00_H$ ... $4FFF_H$ 5900_H ... $5C7F_H$ 6400_H ... $65FF_H$ 7000_H ... $73FF_H$ 8000_H ... $FFFF_H$
TIR	Access is allowed to the flag area and open data blocks only
TNB	For all source bytes: Same restrictions as LIR For all destination bytes: Same restrictions as TIR

Warning

The TIR, TBS and TNB operations are memory-modifying operations which allow you to access user memory and/or the system data area. Improper use of these operations may lead to changes in the user program and to S5-95F malfunctions.

Access to S5-95F System Data

Warning

Note that you may only use the system data bytes listed in Table 6.6. Improper use of system data may result in malfunctioning of the S5-95F, and can thus impair the system's failsafety.

18.13.3 STOP Operation in the User Program

Safety Note

A STOP initiated by the ***"STP" operation*** can very easily be cancelled by means of a programmer input (even inadvertently). A STOP produced by the "STP" operation is therefore ***not a failsafe STOP condition*** (see section 18.16).

18.13.4 Waiting Times in the User Program

Timers T 0 to T 127 are non-retentive. If a power failure occurs while a timer containing a minimum waiting time is still running, the user program must see to it that this waiting period is maintained. To do this, store the pertinent information on the minimum waiting time in a retentive flag, and evaluate that flag in OB22.

18.13.5 Scratchpad

Safety Note

Flags F 200.0 to F255.7 are reserved for the internal programming of integral function blocks and loadable standard FBs. For reasons of safety, the use of these flags in the user program is permitted only when

- no integral function blocks or loadable standard function blocks are used
- the flag area is saved prior to invoking a loadable standard FB or integral function block following execution of that block.

18.13.6 Post-Loading STEP 5 Blocks in Test Mode

The S5-95F permits post-loading of STEP 5 blocks in Test mode. Post-loaded blocks are not processed until the next cycle.

In order that the S5-95F be able to process post-loaded blocks, they must first be compiled, i.e. translated. The amount of time the S5-95F needs to translate a post-loaded block depends on the length of the STEP 5 program in that block, and can be as much as 3 s. Exceeding of the cycle time is therefore a possibility when post-loading long blocks.

18.13.7 Monitoring the Supply Voltage for OB3 Interrupt DI

Safety Note

When there is no supply voltage for the onboard interrupt DIs/counter inputs ***in either subunit, and when no counters have been configured,*** the S5-95F does not flag an error.
If you use the interrupt DI, you must therefore monitor the supply voltage in the user program. To do so, evaluate diagnostic byte IB35 under safety-related conditions.

18.13.8 Triggering Edge for OB3 Interrupts

Safety Note

Execution of the interrupt service routine in OB3 can be initiated by a) a falling edge, b) a rising edge or c) a falling or rising edge at a software interrupt input. The edge(s) to trigger execution of OB3 must be specified with COM 95F.
Always choose a falling edge to trigger an interrupt. You should deviate from this only in very special cases, and only after consulting with the responsible authority.

If the appropriate parameters are assigned, the S5-95F responds to a signal edge change at an interrupt DI as follows:

- On the falling edge as soon as a signal change from 1 to 0 is detected ***in one subunit***.
- On the rising edge when a signal change from 0 to 1 is detected ***in both subunits***.

Special Feature of Interrupt DIs with OB1-Oriented Discrepancy Time

In systems with basic units 095-8FA02 and 095-8FB01 you can also use the interrupt DIs with an OB1-oriented discrepancy time.

To ensure detection of a continuous 1 fault by the S5-95F, the duration of the ”0” and ”1” signal states must be at least: ***parametrized discrepancy time + one OB1 cycle***.

18.13.9 Measures for Securing the User Program Against Errors

In safety mode, the user program must be stored on an EPROM submodule. The EPROM submodules for the two subunits are identical.

Safety Measures for the User Program Provided by the S5-95F

The S5-95F examines the identity and integrity of the programs in subunits A and B in its RESTART routine.

The S5-95F goes to STOP when

- The two subunits' EPROM submodules contain different programs
- The EPROMs were replaced without execution of an overall reset and the EPROMs previously in the subunits had a different system ID number
- Loaded standard function blocks were damaged or manipulated

Safety Measures Against Systematic Programmer Errors

The COM 95F software package contains an acceptance tool with which the user program can be checked for programmer errors. The tool can detect systematic programmer errors and programmer hardware failures (refer to the COM 95F Manual for details).

18.14 Addressing and Address Assignments

The S5-95F's I/Os are subdivided into onboard I/Os and external I/Os.

In order that it be possible to address specific inputs and outputs, the I/Os must be assigned addresses. Please refer to Chapter 6 for details on addressing and address assignments.

The addresses of the onboard I/Os are fixed, and cannot be changed. The addresses of the external I/Os are slot-dependent.

Address Assignment for External I/O Modules

When you mount an external I/O module in a slot on a bus unit, the module is immediately assigned a slot number and consequently a fixed byte address in the process image I/O tables.

Connect the sensors and actuators either via a connector or to the terminal block of the bus units. The terminal selected determines the bit address (channel number) of the input or output.

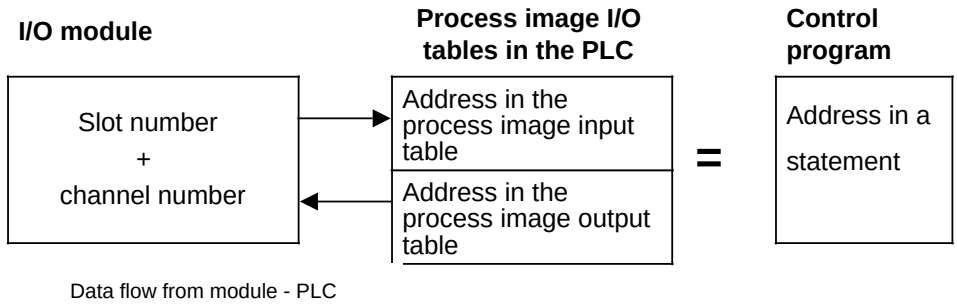


Figure 18-4. Address Assignment

Addressing of Single-Channel External I/Os

Single-channel I/Os are never failsafe. In the user program, these I/Os must always be referenced under the address under which they are used in the relevant subunit.

Addressing of Two-Channel, Failsafe External I/Os

Two-channel I/Os can be used only with failsafe onboard I/Os or with redundant failsafe external I/O modules.

> **Note**
>
> You must always address failsafe external I/Os in the user program at the address where they are used in subunit A (failsafe external I/Os have always an even-numbered byte address)!

18.14.1 Address Assignments for Onboard I/Os

The addresses of the onboard I/Os are permanently assigned and cannot be changed by you. The assignment is shown in the following table:

Onboard I/Os	Byte/Word Address	Bit Address
Digital inputs, failsafe	IB 32 to 33	I 32.0 to I 33.7
Digital outputs, failsafe	QB 32	Q 32.0 to Q 32.7
Digital outputs, non-failsafe	QB 33 to 34	Q33.0 to Q 33.3 (subunit A) Q 34.0 to Q 34.3 (subunit B)
Counter A, failsafe	IW 36	-
Counter B, failsafe	IW 38	-
Interrupt inputs, failsafe	IB 59	I 59.0 to I 59.3

18.14.2 Slot Numbering and Address Assignment for External I/Os

With 8 bus units (16 slots), each subunit can have a maximum of four tiers.

The slots are assigned ascending numbers. Numbering begins with the slot next to the basic unit. Whether a module is plugged in or not has no effect on the numbering.

Note

The slots for subunit A are even-numbered and the slots for subunit B are odd-numbered.

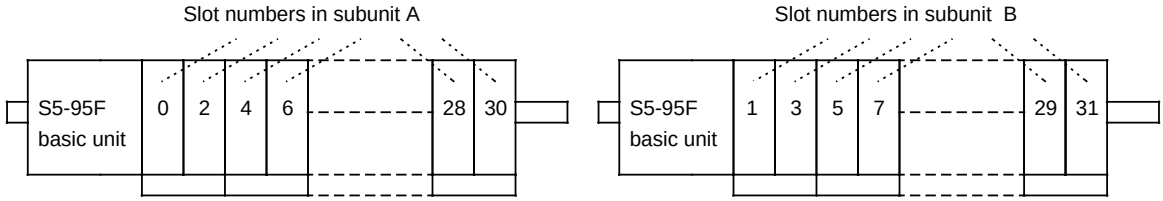


Figure 18-5. Consecutive Numbering of Slots in a Single-Tier Configuration

If the S5-95F consists of several tiers, numbering of the expansion tiers is continued with the slot on the extreme left.

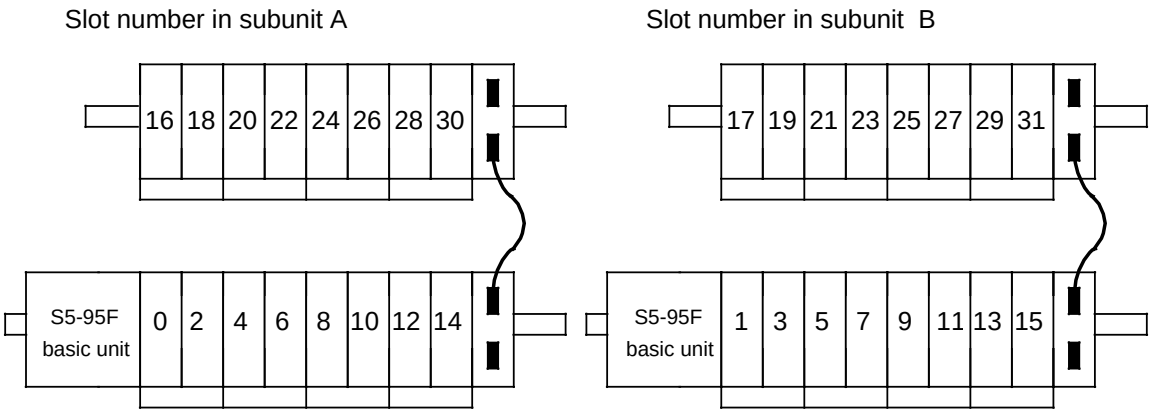


Figure 18-6. Slot Numbering in a Multi-Tier Configuration

When expanding your system, always add the new bus units to the topmost tier on the right. Otherwise, the slot numbers on the right of the new bus units will be changed, which requires address changes in your control program.

Note

After every expansion, check to make certain that the addressing used in the control program is the same as that in the actual configuration.

18.15 Loadable and Integral Function Blocks

Frequently recurring or particularly complex program sections (such as signalling or arithmetic functions) are programmed in function blocks.. These blocks can be assigned parameters and have an extended operation set (for example jump operations within a block).

The following section describes the essential differences and exceptions regarding the approval of function blocks.

Function blocks can be subdivided into the following classes:
- Loadable function blocks, which must be loaded using a programming device
- Integral function blocks, which are part of the S5-95F firmware.

18.15.1 Loadable Function Blocks

The user can program the loadable function blocks himself or use prototype-tested standard function blocks from the SIMATIC library.

Function Blocks Written by the User

Function blocks written by the user must be tested by the inspector during the pre-acceptance/acceptance inspection, as must all other self-programmed STEP 5 blocks.

The test entails at least a discussion about the function, functional tests with viewing of possible error states, and a code anaylsis.

Standard Function Blocks

To help the user and simplify the necessary programming, Siemens offers many prototype-tested standard function blocks available. Catalog ST 50 provides an overview of the standard function blocks available for use in the S5-95F.

The standard function blocks for the S5-95F were tested by independent inspectors. Each block has a unique library number, and is protected by additional measures against falsification due to transfer or disk errors.

Standard function blocks developed for the SIMATIC U-range controllers cannot be used in the S5-95F, and will be rejected by the S5-95F controller (S5-95F will not start).

The standard function blocks are divided into failsafe and reaction-free blocks. Only failsafe standard function blocks may be used to generate safety-related signals.

When assigning parameters to the failsafe blocks, note that an output parameter is failsafe only when all input parameters are also failsafe.

Because the standard function blocks are already prototype-tested and well protected against falsification, the safety test for these blocks has been restricted by the on-site inspector to the following points:

- Comparison of the library number with the library number listed in the expert's report
- Testing of the function block for usage to the intended purpose
- Checking the block call and the FB's parameters as per the block description
- Checking for observance of the conditions listed in the expert's report
- Identity check of the loaded standard function block with the aid of the entry in the system event DB 254 (see section 15.3.2).

Exception for the Scratchpad

> **Note**
>
> When using standard FBs or integral FBs, you may use flag words FW 200 to FW 254 in your control program only if you save the flag area used prior to calling a standard or integral FB and reload it after the FB has executed.

18.15.2 Integral Function Blocks

Several function blocks are already integrated in the S5-95F firmware, and you can call these blocks directly in your control program. It goes without saying that these blocks were also tested by an independent inspector when the S5-95F itself was prototype-tested.

The integral function blocks are divided into failsafe and reaction-free blocks. Only failsafe function blocks may be used to generate safety-related signals.

Safety testing of the integral function blocks by the on-site inspector is normally limited to testing the correctness of the block call and the block parameters as per the block description.

Exception for the Scratchpad

> **Note**
>
> When using standard FBs or integral FBs, you may use flag words FW 200 to FW 254 in your control program only if you save the flag area used prior to calling a standard or integral FB and reload it after the block has executed.

Exception for FB232 and FB233 and Failsafe Analog Value Processing

Failsafe analog value processing is possible when you read in the analog signals with the 464-8MG11 analog module and evaluate them with the aid of integral function blocks FB232 and FB233.

Analog module AI 464-8MG11 has no integral test facility (as does e.g. the DI 431-8FA11). For this reason, you must subject the safety-related analog input modules to a function test at certain intervals. The intervals at which the function test must be carried out depend on the I/O type used.

The Table below shows the I/O types and the intervals at which they must be given a function test.

Table 18-10. Interval for the AI 464-8MG11 Function Test at the User Level

I/O Type	Section	Interval for Function Test
Type R 4.2	Section 11.8.1	30 months
Type R 4.4	Section 11.8.2	5 months
Type R 5.1	Section 11.8.3	3 months
Type R 6.1	Section 11.8.4	2 months
Type R 6.2	Section 11.8.5	12 months
Type R 6.3	Section 11.8.6	1.5 months
Type R 6.4	Section 11.8.7	6 months
Type R 6.5	Section 11.8.8	12 months

Response for Failsafe Analog Value Processing

Safety Note

If there is a discrepancy in the analog signals read, function blocks FB232 and FB233 output a substitute value and flag the error in output parameter DISK. The safety response (e.g. PLC STOP, disconnecting actuators, etc.) must be initiated at the user level.

Exception for FB235 and Connection of Text Displays or Operator Panels

Function block FB235 supports data interchange between the S5-95F and a text display or operator panel connected to the serial interface of a CP 521 SI. The data exchange is handled in free ASCII protocol (FAP).

Non-failsafe data can be entered into a data block via the text display or operator panel while the controller is in failsafe RUN mode. All input data must be checked in the user program for plausibility and non-critical values. The same rules apply as for the use of the parameter control DB (see section 18-16).

18.16 Programmer-Based Operator Input Functions

For reasons of safety, the programmer-based input functions are restricted in safety mode. In this mode, only Read functions may be invoked on the programmer.
The only exceptions are the input DB for FB235 and the parameter control DB, which may be read and modified via programmer, text display or operator panel even in safety mode.

FB235's Input DB

When the controller is in RUN in safety mode, you can modify the input DB specified in FB235 via a text display or operator panel.

Because a data transfer from a text display or operator panel to the S5-95F is only reaction-free, the input data must be treated as non-failsafe data. Before processing the input data in the user program, you must "filter" them for plausibility and safety limits.

The filter program is tested for approval by the inspector in the same way as all other user program blocks.

Parameter Control DB

In safety mode, you can use the programmer function "Output block" to alter the parameter control DB when the PLCs are at STOP. You must specify the parameter control DB during parameter initialization with COM 95F. This DB allows you to specify certain system parameters "during operation".

Before you process these parameters in the user program, however, they must be submitted to a so-called filter routine to check them for plausibility and safety limits.
The authorized inspector must examine and approve the filter routine just as he must all other blocks in the user program.

STOP via Programmer Input Command

> **Safety Note**
>
> There is no interlock feature to prevent you from switching the PLC from STOP to RUN, and you can do so simply by pressing a button on the programmer. For this reason, the STOP initiated via the programmer may not be regarded as safety condition.
> When maintenance is necessary, you should therefore always switch the S5-95F off via its own switch.

Increase in the Response Time Due to Programmer Function

> **Note**
>
> When executing the STATUS function, the S5-95F processes the blocks in a special mode; the blocks' runtimes are thereby considerably increased, and are operation-dependent. Note that the response time is also increased. You should therefore keep your blocks as short as possible.
> The S5-95F monitors the PLC cycle time, and makes sure that the response time is never more than double the maximum PLC cycle time specified with COM 95F.

Programmer Input in Safety, Quasi-Safety and Test Mode

Table 18.11 Programmer Input in Safety, Quasi-Safety and Test Mode

Programmer-Based Functions	Abbr.	Test Mode		Quasi-Safety/ Safety Mode	
		STOP	RUN	STOP	RUN
Inputting/outputting blocks					
Input block DB, FB, OB, PB, SB Parameter control DB	INPUT	 Yes Yes	 Yes Yes	 No Yes	 No No
Output block with editing DB, FB, OB, PB, SB Parameter control DB	INPUT	 Yes Yes	 Yes Yes	 No Yes	 No No
Output block without editing DB, FB, OB, PB, SB	OUTPUT	 Yes	 Yes	 Yes	 Yes
Test					
Signal status display with editing FB, OB, PB, SB	STATUS	 Yes	 Yes	 No	 No
Signal status display without editing FB, OB, PB, SB	STATUS	 Yes	 Yes	 Yes	 Yes
PLC functions					
Start PLC	START	Yes	No	Yes	No
Stop PLC	STOP	No	Yes	No	Yes
Compress PLC memory	COMP	Yes	Yes	Yes	No
Status Variable	STAT VAR	Yes	Yes	Yes	Yes
Control Variable with editing	CTRL VAR	Yes	Yes	No	No
Control Variable without editing	CTRL VAR	Yes	Yes	Yes	Yes
PLC info					
Output address with editing	OUTP ADD	Yes	Yes	No	No
Output address without editing	OUTPADD	Yes	Yes	Yes	Yes
Memory configuration	MEM CONF	Yes	Yes	Yes	Yes
Read system parameters	SYSPAR	Yes	Yes	Yes	Yes
Output block stack	BSTACK	Yes	No	Yes	No
Output interrupt stack	ISTACK	Yes	No	Yes	No
Auxiliary functions					
Transfer block from PG to PLC DB, FB, OB, PB, SB Parameter control DB	TRANS	 Yes Yes	 Yes Yes	 No Yes	 No No
Transfer block from PLC to PG	TRANS	Yes	Yes	Yes	Yes
Overall PLC reset	RESET	Yes	No	Yes	No
Output directory	DIR	Yes	Yes	Yes	Yes

18.17 Connecting Operator Panels and Text Displays

In addition to programmers and SINEC L1 components, failsafe technology also allows the interfacing of text displays and operator panels to the S5-95F via the latter's serial port, as the ports are reaction-free.

Integral function block FB 235, which has been approved and certified as reaction-free, must be used for operator panels connected to the reaction-free interface of a CP 521 SI.

Safety-Related Data Areas Modifiable via Operator Panels and Text Displays

In safety mode, the S5-95F allows text displays and operator panels to modify only

- the parameter control DB while the PLCs are at STOP in safety mode
- the input DB via FB 235 while the PLCs are at RUN in safety mode
- the SINEC L1 Receive mailbox for non-failsafe data transfers

In order to ensure that the data entered in the parameter control DB cannot lead to unsafe system states, this data must be checked for plausibility, validity and range limits in the user program. This check routine must be run before the S5-95F processed this data.

Text displays and operator panels may not be used to transfer failsafe frames to the S5-95F.

Initiating a STOP via Devices Connected to the Serial Interface

> **Safety Note**
>
> A STOP initiated via text displays connected to the serial interface must never be regarded as safe, as the S5-95F could conceivably be switched to RUN via that very device.
> For this reason, switch the S5-95F off via its own switch when doing maintenance.

Conditions for Connecting Operator Panels and Text Displays

Before connecting operator panels or text displays, note the following carefully:

- Operator panels and text displays must be connected using shielded cables. All connections, wiring and cabling must meet all EMC requirements (see section 3.4).
- The operator panels and text displays must be operated using power supply units with protective separation to EN 60950.

No Modification of the User Program via Operator Panels or Text Displays

> **Safety Note**
>
> Modifications to the user program via operator panels or text displays are not permitted.

18.18 SINEC L1 LAN

Note the following when operating a SINEC L1 LAN:

- The SINEC L1 master may be any SIMATIC U-range programmable controller, a CP 541, a PC with master capability, or another device with that capability.
- The SINEC L1 master may not have an interrupt list.
- The SINEC L1 master may not initiate any safety-related actions (such as failsafe STOP or START of a slave).
- A description of the entire data flow must be available for the SINEC L1 LAN. The approval process is simplified e.g. when I/O signals do not travel several SINEC L1 paths in succession.
- COM 530 may not be active in failsafe mode.
  The keylock switch on the associated PLC must be set to LOCK.

A detailed description of SINEC L1 can be found in Chapter 13.

Length of the Receive Mailboxes for Non-Failsafe Data Interchange

Safety Note

Regardless of the intended frame length, the S5-95F's ***Receive mailbox for non-failsafe data interchange*** must

- either have a length of 66 bytes or
- be located at the end of the flag area of data block

Conditions for failsafe input/output signals

Safety Note

If failsafe input/output signals are to be transmitted over the SINEC L1 network, a "0" signal must always result in a safe system state (***quiescent current principle***). This condition must be fulfilled because the S5-95F erases the contents of the Receive mailboxes in the event of a data transmission error.

Inputs are in a safe state when the transmitted "0" signal brings the process to a safe quiescent state. The input for an EMERGENCY STOP, for instance, must have a "1" signal during operation and be activated by a "0" signal.

An output is in a safe state when the transmitted "0" signal resets the output and brings the actuator to a safe state.

Coordinating Frame Sequences (95F Mode)

The S5-95F does not send a "new" useful message frame until the receiver's operating system has retrieved the data in the "old" useful message frame. The transmitter, however, does not receive an acknowledgement which tells it whether the receiver's user program has actually evaluated the useful message frame.

Because the S5-95F clears the Receive mailbox when a power failure occurs, frame loss ensues the receiver's operating system received the frame but the user program did not evaluate it before the power failed. If your automated process cannot tolerate frame loss, you must monitor the frame sequence at the user level via an acknowledgement system. This requires a data path from the receiver to the transmitter.

Safety Note

SINEC L1 nodes receive no acknowledgement to inform them whether the receiver has actually evaluated the frame that was sent. If your automated process requires failsafe frame coordination, you must monitor the frame sequence with an acknowledgement system.

Coordinating Frame Sequences (115F-14 and 115F-15 Modes)

The S5-95F resets bit 0 or 2 of the UVB after sending the frame, even if the receiver's operating system has not accepted the data of the old frame.

Safety Note

When you use message modes 115F-14 and 115F-15, the S5-95F useful frame monitoring feature is deactivated. To prevent an undetected frame loss, you must not alter the send mailbox more than once during each safety time for Receive. If your automated process requires failsafe frame coordination, you must monitor the frame sequence with an acknowledgement system.

Restrictions When Using the S5-95F with Basic Units 095-8FA01

You specify the system response to SINEC L1 errors when assigning the parameters with COM 95F.



Warning

The parameter entry **"Slave type S5-115F" with "Response to SINEC L1 error in user program"** is ***not permitted*** for S5-95F PLCs with basic units 095-8FA01. This parameter assignment could result in a ***nonfailsafe response*** from the S5-95F.

18.19 SINEC L1 Safety Times

In a failsafe SINEC L1 system, the S5-95F differentiates between two types of safety times, i.e. the SINEC L1 safety time for Receive and the SINEC L1 safety time for Send. The safety times may be set separately for each data path (max. four different safety times).

Problems with the SINEC L1 Bus Master

When the SINEC L1 bus master is not functioning properly, e.g. because it is at STOP or because of a restart following a POWER OFF/POWER ON sequence, failsafe frames might be delayed. If a delay is excessive, the SINEC L1's watchdog timer containing the safety time for Receive could expire (timeout). In this case, the S5-95F responds by going to STOP or by passivating the data path, depending on what was specified in DB1.

18.19.1 SINEC L1 Safety Time for Receive

Definition: The SINEC L1 safety time for Receive stipulates a time period for failsafe data paths during which the receiving node must have received at least one valid frame.

The SINEC L1 safety time for Receive depends on the automated process and on the user programs of the participating partners in the communications link. The value chosen as safety time for Receive must be approved by the authorized inspector.

Discuss with the inspector the following points for each failsafe data path:

- The SINEC L1 safety time for Receive
- The system response when no valid frame is received within the time period stipulated as SINEC L1 safety time for Receive.

> **Note**
>
> It is not necessary to send and receive at least one useful data frame within the SINEC L1 safety time. If necessary, the operating system transmits additional test frames, unbeknownst to the user, for the express purpose of monitoring the bus.

Interval for Receiving Frames

> **Note**
>
> Please note that the S5-95F cannot receive a new frame until it has processed the previous one. You should therefore arrange communications between nodes so that frames are received at least 100 ms apart, otherwise the S5-95F may not accept the next frame.

Conditions for the SINEC L1 Safety Time for Receive

1st condition

The following condition requires consideration of the transmitting and the receiving system.

The receiving S5-95F must fulfill the following as regards the SINEC L1 safety time:

$$\text{SINEC L1 safety time for Receive} \quad \frac{2\cdot(k+l)\cdot\text{SINEC L1 bus cycle time}}{n} + 200\ \text{ms}$$

k= Number of failsafe Send paths used by the S5-95F (k=1 or 2) or S5-115F (k=1 to 30), where:
- k=1 for one failsafe data path
- k=2 to 30 for 2 to 30 failsafe data paths (the data path to the master must also be included in the case of the S5-115F)

l= Factor for non-failsafe data traffic, where
- l=0, when the relevant transmitter sends non-failsafe data only rarely or not at all
- l=1, when the relevant transmitter sends non-failsafe data frequently

n= Number of times the relevant transmitter is polled in the polling list
If polled more than once, the node number must be evenly distributed throughout the list, e.g.:
1 **2** 3 4 **2** 5 6 **2** 7 8 **2** 9

SINEC L1 bus cycle time =
- Number of frame bytes (plus 4 bytes in the case of failsafe frames)×2 ms
- \+ Number of failsafe data paths in the entire SINEC L1 network×44 ms

TIP: You can reduce the SINEC L1 bus cycle time by specifying smaller amounts of data and/or fewer data paths in the polling list.

2nd condition

The time between two successive frames from the master to a given slave must be 1.5 x slave's SINEC L1 safety time for Receive.

3rd condition (for double SINEC L1 bus only)

When a given S5-95F receives frames from the master and a double SINEC L1 bus is being used, the following condition must be fulfilled:

SINEC L1 safety time for Receive SINEC L1 safety time for Send + 500 ms

The formula for computing the SINEC L1 safety time for Send can be found in section 18.19.2.

Error Response from Slave Node

If a node fails to receive a valid frame within the stipulated SINEC L1 safety time for Receive,

- The operating system sets bit 0 in the "Condition code and control byte for Receive" and clears the Receive mailbox
- The operating system flags an error in the SINEC L1 error word in system event data block DB254
- The operating system enters an error message in the error buffer in system event data block DB254
- The S5-95F responds as specified in DB1 (possible responses are a PLC STOP or the reaction written in the user program)

18.19.2 SINEC L1 Safety Time for Send

The S5-95F controls transmission of failsafe frames via an internal timer. The S5-95F restarts the timer as soon as it expires. As long as the internal timer is running, the S5-95F transmits (during one timer cycle) the failsafe frames over data paths 1 and 2; then, if there is enough time left, it transmits non-failsafe frames.

The S5-95F computes the initial value for the internal timer from the lower of the two SINEC L1 safety time for Send values (data path 1 and data path 2). The formula for the initial internal timer value is:

Internal timer = 0.5 x SINEC L1 safety time for Send - 100 ms

Conditions for the SINEC L1 Safety Time for Send

The SINEC L1 time for send must fulfil the following conditions:

1st condition

$$\text{SINEC L1 safety time for Send} \quad \frac{2\cdot(k+l)\cdot\text{SINEC L1 bus cycle time}}{n}$$

k= Specified number of failsafe data paths for transmitting, where
- k=1 for one failsafe data path
- k=2 for two failsafe data paths

l= Factor for non-failsafe data paths, where
- l=0, when non-failsafe data is transmitted only rarely or not at all
- l=1, when non-failsafe data is transmitted frequently

n= Number of times the S5-95F (node number of transmitter) is polled in the SINEC L1 polling list
If the S5-95F is polled more than once, its node number must be evenly distributed throughout the list, e.g.: 1 **2** 3 4 **2** 5 6 **2** 7 8 **2** 9

SINEC L1 bus cycle time = Number of frame bytes (plus 4 bytes for failsafe frames)×2 ms
\+ Number of failsafe data paths in the entire SINEC L1 network×44 ms

2nd condition

The SINEC L1 safety time for Send must be 300 ms

3rd condition (double SINEC L1 bus only)

This condition must be fulfilled when using a double SINEC L1 bus:

SINEC L1 safety time for Receive SINEC L1 safety time for Send + 500 ms

4th condition (for single SINEC L1 bus only)

This condition must be fulfilled when using a single SINEC L1 bus:

SINEC L1 safety time for Receive SINEC L1 safety time for Send + 200 ms

18.19.3 Response Time During SINEC L1 Traffic

The response time during SINEC L1 traffic is the time between the change in the input signal in the transmitting system and the change in the output signal in the receiving system.

The response time is computed from three different times:

- The processing time in the transmitter (time between change in the input signal and updating of the SINEC DB, including Send request in User Valid Byte)
- SINEC L1 safety time for Receive
- Processing time in the receiver (time between evaluation of the SINEC L1 DB, including Receive request in User Valid Byte, and the change in the output signal)

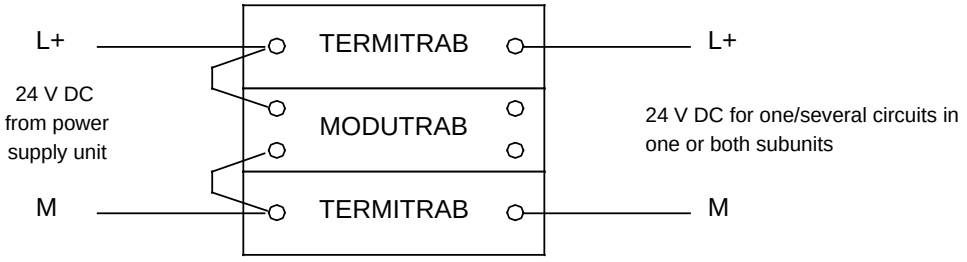
18.20 Filter for 24 V DC Power Supply Units

In order that the S5-95F fully satisfy the requirements to IEC 801-5 (surge pulse), Severity Level III, the 24 V DC voltage for the following components must be filtered:

- Onboard interrupt DIs/counter inputs
- Digital input module DI 431-8FA11
- Digital output module DQ 450-8FA11

Filter for Compliance with the Safety Regulations to IEC 801-5, Severity Level III

Run the 24 V DC voltages generated by power supply units for the above-listed components through a filter. Use the circuitry shown in the Figure with the specified components.



To reduce parasitic signals, the lines between filter and S5-95F may not exceed 50 cm in length

Figure 18-7. Filter for Safety Regulations to IEC 801-5, Severity Level III

Table 18-12. Components for Mains Filters

Component/Type	Manufacturer	Order No.
TERMITRAB Type SLKK-S/60 AC	PHÖNIX-CONTACT	27 94 974
MODUTRAB Type MT-2/1-S-24 DC	PHÖNIX-CONTACT	27 65 699

18.21 EMC of the IM 316-8MA12

Safety Note

For reasons of electromagnetic compatibility, unused sub D slots in the IM 316-8MA12 must be covered. To do so, use a blank sub D connector (6ES5-750-2AA21).

Appendices

Appendix A	Module Spectrum
Appendix B	Dimension Drawings
Appendix C	Operations List
Appendix D	Guidelines for Handling Electrostatic Sensitive Devices (ESD)
Appendix E	Prototype Test Certification

A Module Spectrum

A.1	General Technical Specifications for Failsafe Modules	A - 4
A.1.1	Programmable Controller	A - 5
A.1.2	Digital Modules	A - 7
A.1.3	Analog Modules	A - 9
A.1.4	Bus Units	A - 11
A.1.5	Interface Modules	A - 13
A.2	Standard Modules from the S5-100U Range	A - 14
A.2.1	Digital Input Modules	A - 16
A.2.2	Digital Output Modules	A - 23
A.2.3	Digital Input/Output Modules	A - 31
A.2.4	Analog Input Modules	A - 33
A.2.5	Analog Output Modules	A - 66
A.2.6	Function Modules	A - 75
A.2.7	Bus Units	A -106
A.2.8	Interface Modules	A -108

Figures

A-1 Connecting Thermocouples of the Same Type with Linearization and Internal Compensation Directly to the Module . . . . . A - 37
A-2 Remote Connection of Thermocouples of the Same Type with Linearization and Internal Compensation to the Module . . . . . A - 37
A-3 Connecting Thermocouples of the Same Type without Linearization and External Compensation Directly to the Module . . . . . A - 38
A-4 Remote Connection of Thermocouples of the Same Type without Linearization and External Compensation to the Module . . . . . A - 38
A-5 Direct and Remote Connection of Thermocouples of Different Type without Linearization and External Compensation to the Module . . . . . A - 39
A-6 Connection of only Four-Wire Transducers (6ES5 464-8MD11) . . . . . A - 56
A-7 Connection of only Two-Wire Transducers (6ES5 464-8MD11) . . . . . A - 56
A-8 Connection of Two-Wire and Four-Wire Transducers (6ES5 464-8MD11) . . A - 57
A-9 Connection of only Two-Wire Transducers (6ES5 464-8ME11) . . . . . A - 62
A-10 Connection of only Four-Wire Transducers (6ES5 464-8ME11) . . . . . A - 62
A-11 Connection of Two-Wire and Four-Wire Transducers (6ES5 464-8ME11) . . A - 63
A-12 Load Connection via a Four-Wire Circuit (6ES5 470-8MA12) . . . . . A - 68
A-13 Connection via a Two-Wire Circuit (6ES5 470-8MB12) . . . . . A - 70
A-14 Connection via a Two-Wire Circuit (6ES5 470-8MC12) . . . . . A - 72
A-15 Load Connection via a Four-Wire Circuit (6ES5 470-8MD12) . . . . . A - 74
A-16 Positioning with the IP 263 . . . . . A - 82
A-17 Processed Units of Measurement for Circular Axis and Linear Axis . . . . . A - 87
A-18 Course of the Following Error during a Positioning Operation . . . . . A - 88
A-19 Velocity Profile of the IP 267 . . . . . A - 91
A-20 Switch Positions on the Operating Mode Switch . . . . . A - 96
A-21 Scanning the Comparator Module . . . . . A - 98

Tables

A-1 Representation of an Analog Input Value as Bit Pattern . . . . . A - 33
A-2 Options for Connecting Thermocouples . . . . . A - 36
A-3 Settings of the Operating Mode Switch for Analog Input Module 464-8MA21 . . . . . A - 41
A-4 Analog Input Module 464-8MA21, 4 x ±50 mV, without Linearization and without Temperature Compensation (Internal) (Bipolar Fixed-Point Number) . . . . . A - 42
A-5 Analog Input Module 464-8MA21, 4 x ±50 mV with Linearization and with Temperature Compensation (Internal); Thermocouple Type K (Nickel-Chromium/Nickel-Aluminum, according to DIN IEC 584) . . . . . A - 43
A-6 Analog Input Module 464-8MA21, 4 x ±50 mV with Linearization and with Temperature Compensation (Internal); Thermocouple Type J (Iron/Copper-Nickel (Constantan), according to DIN IEC 584) . . . . . A - 44

Tables

A-7 Analog Input Module 464-8MA21, 4 x ±50 mV with Linearization and with Temperature Compensation (Internal); Thermocouple Type L (Iron/Copper-Nickel (Constantan), according to DIN 43710) . . . . . A - 45
A-8 Settings for the Operating Mode Switch for Analog Input Module 464-8MB11 . . . . . A - 49
A-9 Analog Input Module 464-8MB11 (Bipolar Fixed-Point Number) . . . . . A - 49
A-10 Settings for the Operating Mode Switch for Analog Input Module 464-8MC11 . . . . . A - 52
A-11 Analog Input Module 464-8MC11 (Bipolar Fixed-Point Number) . . . . . A - 53
A-12 Settings for the Operating Mode Switch for Analog Input Module 464-8MD11 . . . . . A - 58
A-13 Analog Input Module 464-8MD11 (Bipolar Fixed-Point Number) . . . . . A - 59
A-14 Settings of the Operating Mode Switch for Analog Input Module 464-8ME11 . . . . . A - 64
A-15 Analog Input Module 464-8ME11, 4 x 4 to 20 mA (Absolute Value) . . . . . A - 65
A-16 Representation of an Analog Output Value as Bit Pattern . . . . . A - 66
A-17 Analog Output Module 470-8MA12 (Bipolar Fixed-Point Number) . . . . . A - 68
A-18 Analog Output Module 470-8MB12 (Bipolar Fixed-Point Number) . . . . . A - 70
A-19 Analog Output Module 470-8MC12 (Unipolar) . . . . . A - 72
A-20 Analog Output Module 470-8MD12 (Unipolar) . . . . . A - 74
A-21 Designation of the Operating Modes . . . . . A - 89

A Module Spectrum

Information on the CE Marking for the SIMATIC S5

Our products meet the requirements and protection guidelines of the following EC Directives and comply with the harmonized European standards (EN) issued in the Official Journal of the European Communities with regard to programmable controllers:

- 89/336/EEC "Electromagnetic Compatibility" (EMC Directive)
- 72/23/EEC "Electrical Equipment Designed for Use between Certain Voltage Limits" (Low-Voltage Directive)

The declarations of conformity are held at the address below, where they can be obtained if and when required by the respective authorities:
Siemens Aktiengesellschaft
Bereich Automatisierungstechnik
AUT E 14
Postfach 1963
D-92209 Amberg
Federal Republic of Germany

Application

SIMATIC products have been designed for use in the industrial area and meet the following requirements:

Area of Application	Requirements on Emitted Interference	Requirements on Immunity
Industry	EN 50081-2 : 1993	EN 50082-2 : 1995

SIMATIC products can also be used in the domestic environment (household, business and trade area, small plants) with individual approval.

Area of Application	Requirements on Emitted Interference	Requirements on Immunity
Domestic	Individual approval	EN 50082-1 : 1992

You must acquire the individual approval from the respective national authority or testing body. In Germany, individual approvals are granted by the Federal Authority for Post and Telecommunications and its branch offices.

Observing the Installation Guidelines

SIMATIC products meet the requirements if you

1. Observe the installation guidelines described in the manual when installing and operating the equipment.
2. Observe, in addition, the following rules regarding installing the devices and working on switching cabinets and the notes on the individual modules.

Controller Installation

Programmable controllers of the SIMATIC S5-90U, S5-95U/F and S5-100U range must be installed in electrical equipment rooms in closed housings (for example, metal or plastic switchboxes).

Working on Switching Cabinets

In order to protect the modules against static electrical discharge, operating personnel discharge themselves of static electricity before opening switching cabinets or switchboxes.

Information on the Individual Modules

Additional measures are required when using the following modules in an S5-95F.

Order Number	Module	Measures Required
6ES5 266-8MA11	Positioning module	When using the IP 266, the controller must be installed in a grounded, closed metal housing.
6ES5 430-8MB11	Digital input module	When using the DI 430, the controller must be installed in a grounded, closed metal housing. Signal lines must be shielded. Apply the shield of the signal lines on the shielding bar at the entrance to the cabinet.
6EW1 380-4AB01	Load current supply module	A filter must be installed in the supply cable (SIFI C, B84113-C-B30 or equivalent).

Updated Technical Specifications for Standard S5-100U-Series Modules

Contrary to the information in the "General Technical Specifications" of the manual, the EMC specifications listed below apply for modules bearing the CE marking.

The specifications are only valid for devices installed in accordance with the installation guidelines named above.

Electromagnetic Compatibility Specifications	Test Values
Immunity to static electrical discharge tested in accordance with EN 61000-4-2	 Discharge to air 8 kV Discharge to contact 4 kV
Immunity to electromagnetic fields tested in accordance with EN V 50140 (amplitude-modulated HF)	 800 to 1000 MHz 10 V/m 80 % AM (1 kHz)
tested in accordance with EN V 50204 (pulse-modulated HF)	900 MHz 10 V/m 50 % ED, 200 Hz repetition frequency
Immunity to fast transient bursts tested in accordance with EN 61000-4-4 Supply lines for 120/230 V AC Supply lines for 24 V DC Signal lines (I/O and bus lines)	 2 kV 2 kV 2 kV*
Immunity to high frequency tested in accordance with EN V 50141	0.15 to 80 MHz 10 V 80 % AM (1 kHz) Source impedance 150
Emitted interference tested in accordance with EN 55011 Emission of electromagnetic fields Emitted interference over supply cable	 Limit value class A, Group 1 Limit value class A, Group 1

* Signal lines that are not used for process control, for example, connections to external printers: 1 kV

A.1 General Technical Specifications for Failsafe Modules

Climatic Environmental Conditions	
Temperature	
Operating	
- horizontal design	0 to+60 °C (32 to 104 °F)
- vertical design	0 to+40 °C (32 to 104 °F) (Air-intake temperature, measured on the underside of the module)
Storage/transport	- 40 to+70 °C (-40 to 150 °F)
Temperature change	
- operating	max. 10 °C/h (50 °F/h)
- storage/transport	max. 20 °C/h (68 °F/h)
Relative humidity	to DIN 40040 15 to 95 % (indoor), non-condensing
Atmospheric pressure	
- operating	860 to 1060 hPa
- storage/transport	660 to 1060 hPa
Pollutants	
- SO_2	0.5 ppm, (rel. humidity 60 %, non-condensing)
- H_2S	0.1 ppm, (rel. humidity 60, non-condensing)

Mechanical Environmental Conditions	
Vibration	to IEC 68-2-6
- tested with	10 to 150 Hz, (const. ampl. 0.15 mm)
	57 to 150 Hz, (const. accel. 2 g)
Shock	to VDE 0116
- tested with	18 shocks (semisinusoidal 30 g/18 ms)
Free-fall	to IEC 68-2-31
- tested with	height of fall 50 mm

Emitted Electromagnetic Interference	
Radio interference level to EN 55011	Limit class A

Electromagnetic Compatibility (EMC) Noise Immunity	
Electrostatic discharge	to IEC 801-2 severity rating 3
- Test voltage	8 kV air discharge 6 kV contact discharge

Electromagnetic Compatibility (EMC) Noise Immunity	
Rated electromagnetic field test	to IEC 801-3, severity rating 3 field strength 10 V/m
Fast transient burst	to IEC 801-4, severity rating 3
Power supply lines	
- Supply voltage 24 V DC	2 kV
Signal lines	
- Analog input/output modules	1 kV
- Digital input/output modules for V 24 V	2 kV
Communications interfaces	1 kV
Surge impulse	to IEC 801-5, severity rating 3
HF current	to IEC 801-6, severity rating 3

IEC/VDE Safety Information	
Degree of protection	to IEC 529
- Type	IP 20
- Class	I to IEC 536
Insulation rating	to VDE 0160
- between electrically independent circuits **and** with circuits connected to a central grounding point	(05. 1988) to VDE 0160
- between all circuits **and** a central grounding point (standard mounting rail)	(05. 1988) to VDE 0160 (05. 1988)
Test voltage for a rated voltage V_{input} of the circuits (AC/DC)	Sine, 50 Hz
V_{input}=0 to 50 V	500 V
V_{input}=50 to 125 V	1250 V
V_{input}=125 to 250 V	1500 V

A.1.1 Programmable Controller

Technical Specifications AG S5-95F (6ES5 095-8FB01)

Dimensions and Weight

Dimensions WxHxD	(in mm)	145×135×146
	(in in.)	5.7 × 5.3 × 5.7
Weight - S5-95F	approx.	1.5 kg (3.3 lbs.)

Internal Technical Specifications

Memory capacity		
- internal RAM		16 KB, 8 KB for control program
- EPROM submodule		
Clock		
- accuracy t_g		±1 s/day at 25 °C (77 °F)
- variation due to temp. change	t_A	0 to 60 °C (32 to 140 °F)
Execution time		
- per binary operation	approx.	1.2 µs
Cycle monitoring time		
- defined via hardware	approx.	680 ms
- programmable via software		10 to 2550 ms
Flags		2048 (512 retentive)
Timers number/range 128;		0.01 to 9990 s
Counters number/range 128;		0 to 999 (8 retentive)
Inputs/outputs (onboard)		
Digital inputs, failsafe		16
Digital outputs		
- failsafe		8
- single-channel		2×4
Interrupt inputs, failsafe		4
Counter inputs, failsafe		2
External I/O modules		
Digital modules	max.	32
Analog modules	max.	8
Permissible blocks		
Organization blocks		1, 2, 3, 13, 21, 22, 31, 34, 37, 251
Program blocks		0 to 255
Function blocks		0 to 255
- integrated		240 to 243, 250 to 252, and 255
Sequence blocks		0 to 255
Data blocks		1 to 255
- reserved		1, 252 to 255
Operation set	approx.	140

Power Supply (Internal)

Input voltage		
- rated value		24 V DC
- permissible range		20 to 30 V
Current consumption from 24 V		
- for the subunit	typ.	160 mA
- for full external I/O configuration	typ.	1 A
Output voltage		
- U 1 (for external I/O)		+9 V
- U 2 (for programmer)		+5.2 V
Output current		
- from V 1		1 A
- from V 2		0,65 A
Short-circuit protection		electronic
Class of protection		Class I
Floating		no
Back-up battery		Lithium battery (3.4 V/850 mAh)
- life expectancy	min.	1 year
- service life (at 25 °C/77° F)	approx.	5 years
Power loss of the module	typ.	12 W

Specific Onboard I/O Specifications

For digital inputs:		
Floating		yes
- isolated in groups of		16
- Isolation voltage	max.	60 V DC against shield
Input voltage		
- rated value		24 V
- at "0" signal		- 30 V to +5 V DC
- at "1" signal		+13 V to +30 V DC
Input current		
- at "0" signal		<1.5 mA (Input zero-signal current, 2-wire BERO)
- at "1" signal		<6.5 mA (at 30 V)
Delay time		
- from "0" to "1"	typ.	2.5 ms
- from "1" to "0"	typ.	2.0 ms
Cable length unshielded	max.	200 m (660 ft.)

For digital outputs		
Floating	yes	
- isolated in groups of		16
- Isolation voltage	max.	60 V DC against shield
Load voltage L+		
- rated value		24 V DC
- permissible range (including ripple)		20 to 30 V DC
Output current for "1" signal		
in horiz. mounting position at 0 to 40° C with L + 24 V DC		
- failsafe outputs	max.	2.0 A
- single-channel outputs	max.	0.5 A
in horiz. mounting position at 0 to 40° C with L + 30 V DC		
- failsafe outputs	max.	2.0 A
- single-channel outputs	max.	0.5 A
in horiz. mounting position at 40 to 60° C with L + 24 V DC		
- failsafe outputs	max.	1.5 A
- single-channel outputs	max.	0.5 A
in horiz. mounting position at 40 to 60° C with L + 30 V DC		
- failsafe outputs	max.	1.0 A
- single-channel outputs	max.	0.5 A
in vert. mounting position at 0 to 40° C with L + 30 V DC		
- failsafe outputs	max.	1.0 A
- single-channel outputs	max.	0.5 A

For digital outputs (continued):

Total current (simultaneity factor)		
- failsafe outputs		100 %
- single-channel outputs		100 %
Lamp load		
- failsafe outputs		
- source output	max.	4 W
- sink output	max.	10 W
- single-channel outputs	max.	10 W
Residual current for "0" signal	max.	50 µA
Output voltage		
- for "0" signal	max.	0.5 V (at 6k ohms load resistance)
- for "1" signal	max.	L+ minus 0.5 V
	max.	M plus 0.5 V
Short-circuit protection for failsafe outputs		
- source output		electronic
- sink output		thermoelectronic
Short-circuit protection for		
- single-channel outputs		thermoelectronic
Overload withstand capability		
- failsafe outputs		no
- single-channel outputs		no
Polarity reversal protection		
- for failsafe outputs		no
- for single-channel outputs		no
Restriction of the inductive circuit interruption (internal)		
- for failsafe outputs	to	- 11 V
- for single-channel outputs	to	- 16 V
Switching frequency		
- resistive load	max.	100 Hz
- inductive load	max.	2 Hz
Direct digital output control		possible
Paralleling of outputs		
- failsafe outputs		possible when outputs are de-coupled by a diode
- single-channel outputs	each	0.5 A
Cable length, unshielded	max.	200 m (660 ft.)
Current consumption from 24 V		
- failsafe outputs (all outputs "0")		50 mA
- single-channel outputs (all outputs "0")		20 mA

Counter inputs

Floating		yes
Input voltages and currents as for digital inputs (see Specific Onboard I/O Specifications)		
Delay time		
- from "0" to "1"	typ.	1 ms
- from "1" to "0"	typ.	1 ms
Counter frequency*		
- counter A:	max.	900 Hz
- counter B:	max.	900 Hz
Pulse duration		1.4 ms
Cable length (shielded)		100 m

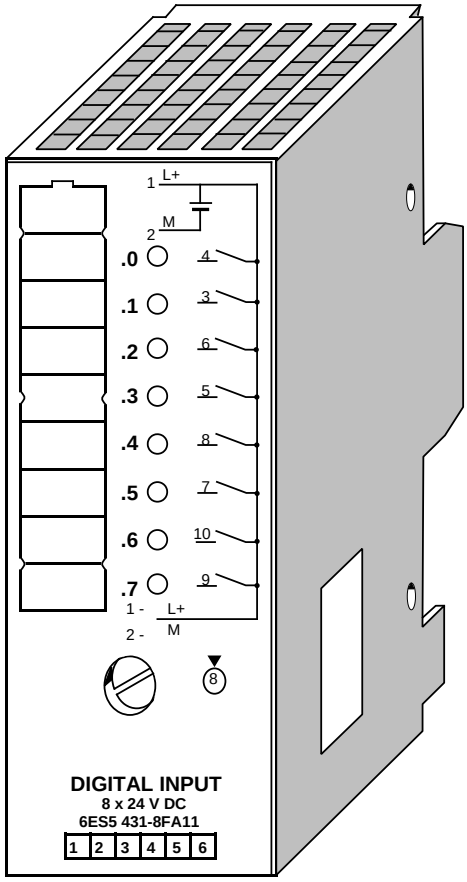
For interrupt inputs

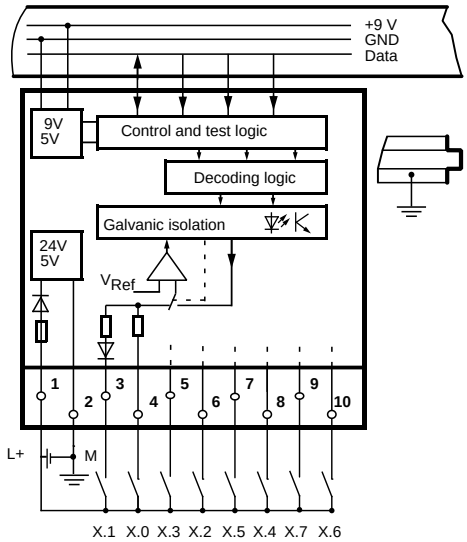
Floating		yes
Input voltages and currents as for digital inputs (see Specific Onboard I/O Specifications)		
Delay time		
- from "0" to "1"	typ.	1 ms
- from "1" to "0"	typ.	1 ms
Pulse duration*		2 ms
Cable length (shielded)		100 m (330 ft.)

* without connection of a programmer, operator panel, text display or SINEC L1

A.1.2 Digital Modules

Digital Input Module 8×24 V DC (6ES5 431-8FA11)
- Safety-Related -

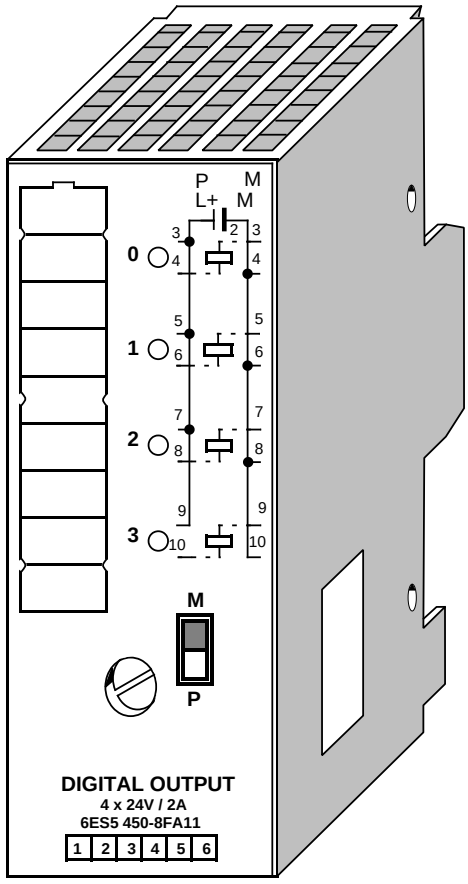


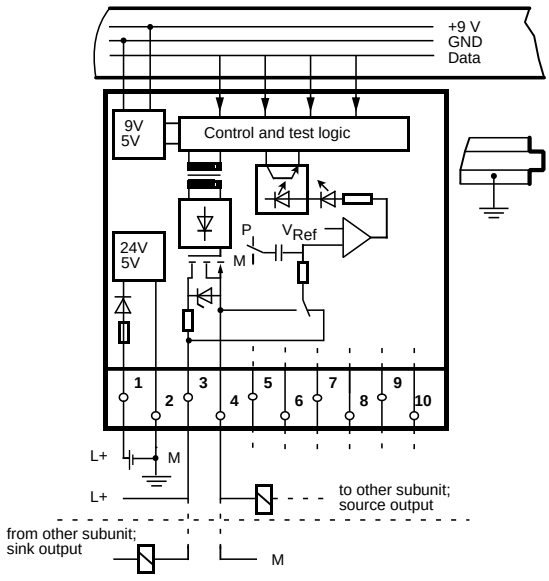


Technical Specifications

Address designation (for ET 100U only)		8 DI
Number of inputs		8
Galvanic isolation		yes (optocoupler)
- in groups of		8
Input voltage L+		
- rated value		+24 V DC
- "0" signal		-30 to+5 V DC
- "1" signal		+13 to+30 V DC
Input current at "1" signal	typ.	9 mA (at 24 V)
Inherent delay		
- from "0" to "1"	typ.	3 ms
- from "1" to "0"	typ.	3 ms
Length of cable		
- unshielded	max.	200 m (660 ft.)
Insulation rating		VDE 0160
Rated insulation voltage (+9 V to L+)		30 V AC
- insulation group		2×B
- tested with		500 V AC
Rated insulation voltage (+9 V to ⏚)		12 V AC
- insulation group		2×B
- tested with		500 V AC
Permissible ambient temperature of PLC		
- horizontal arrangement		0 to 60 °C (32 to 140 °F)
- vertical arrangement		0 to 40 °C (32 to 104 °F)
Connection of 2-wire BERO proximity switches		possible
- residual current		1.5 mA
Current consumption		
- from+9 V (CPU)	max.	70 mA
- from L+	max.	50 mA
Power loss of the module	typ.	2.6 W
Weight	approx.	210 g (7.4 oz.)

Digital Output Module 4×24 V DC/2 A **(6ES5 450-8FA11)**
Source or Sink Output - Safety-Related - **(6ES5 450-8FA12)**



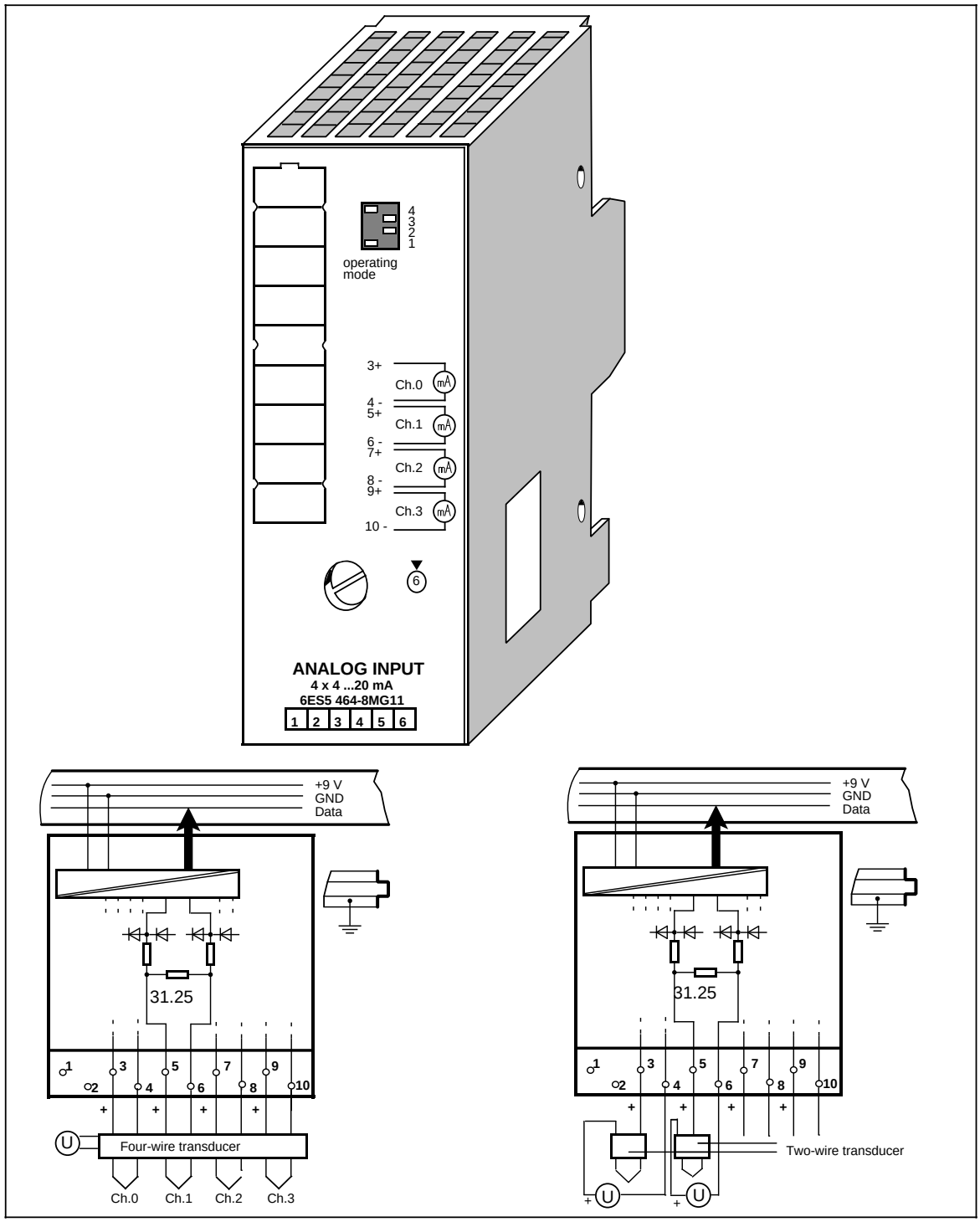


Technical Specifications

Number of outputs		4
Galvanic isolation		yes (transformer)
- in groups of		4
Load voltage L+		
- rated value		24 V DC
- permissible range		20 to 30 V
Output current for "1" signal		
- rated value		2 A at 40 °C/104 °F 1.5 A at 60 °C/140 °F
- permissible range		5 mA to 2 A
- lamp load	max.	20 W
Residual current at "0" signal	max.	1 mA
Short-circuit protection		
- thermal		
- electronic		
Voltage induced on circuit interruption (internal) limited to		- 36 V
Switching frequency		
- resistive load	max.	100 Hz
- inductive load (2 mH)	max.	2 Hz
Total permissible current of outputs		at 40 °C/104 °F: 8 A at 60 °C/140 °F: 6 A
Driving of digital input		possible
Parallelling of outputs		possible, if outputs decoupled by diodes
Permissible ambient temperature of PLC		
- horizontal arrangement		0 to 60 °C (32 to 140 °F)
- vertical arrangement		0 to 40 °C (32 to 104 °F)
Length of cable		
- unshielded	max.	200 m (660 ft.)
Insulation rating		to VDE 0160
Rated insulation voltage (+9 V to L+)		60 V AC
- Insulation group		2×B
- tested with		500 V AC
Rated insulation voltage (+9 V to ⏚)		12 V AC
- insulation group		1×B
- tested with		500 V AC
Current consumption		
- from+9 V (CPU)	max.	60 mA
- from L+(without load)	max.	36 mA
Power loss of the module	typ.	5 W
Weight	approx.	325 g (11.4 oz.)

A.1.3 Analog Modules

Analog Input Module 4 x 4 to 20 mA **(6ES5 464-8MG11)**



Analog Input Module 4 x 4 to 20 mA (continued) (6ES5 464-8MG11)

Technical Specifications

Input ranges (rated values)		4 to 20 mA
Inputs		1, 2 or 4 (switchable)
Galvanic isolation		
- inputs against ground		yes
- inputs against each other		no
Input resistance		31.25 ohms
Signal sensor connection		two-wire connection for two-wire and four-wire transducers
Digital representation of input signal		12 bits + sign (2048 units = rated value)
Measured-value representation		two's complement (left justified)
Measuring principle		integrating
Conversion principle		voltage-time conversion (dual slope)
Integration time (specifiable for optimum parasitic voltage suppression)		20 ms at 50 Hz 16.6 ms at 60 Hz
Encoding time per input		
- at 2048 units	max.	60 ms at 50 Hz
	max.	50 ms at 60 Hz
- at 4095 units	max.	80 ms at 50 Hz
	max.	66.6 ms at 60 Hz
Permissible potential difference		
- inputs to each other	max.	±1 V
- inputs to central earthing point	max.	75 V DC/60 V AC
Permissible input current (destruction limit)	max.	80 mA
Error flagged on		
- range violation		yes (over 4095 units)
- wirebreak in signal sensor lines		no
- wirebreak group flag		no

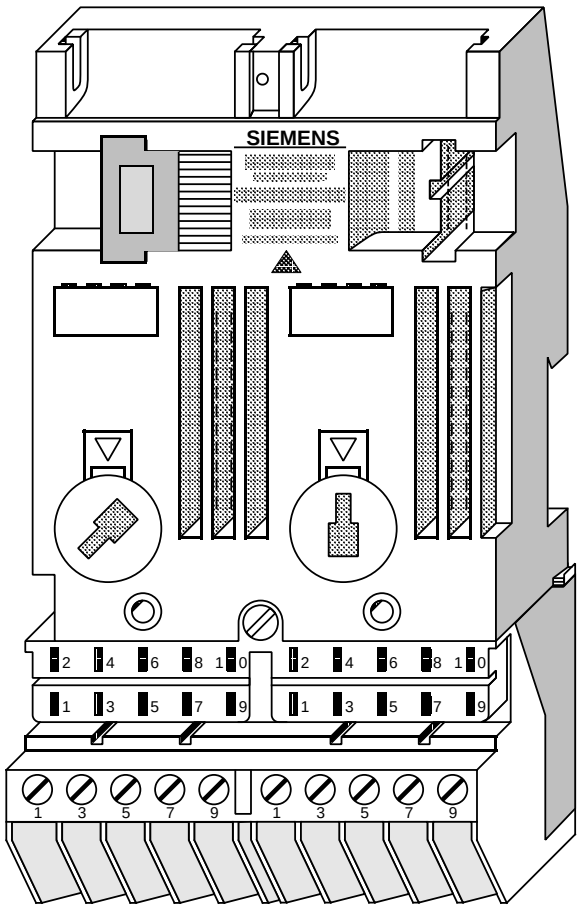
Parasitic voltage suppression for f=nx (50/60 Hz±1 %); n=1, 2, ...		
- common mode interference (V_{ss}=1 V)	min.	86 dB
- series mode interference (peak value of interference < rated value of input range)	min.	40 dB
Basic error (operational limit at 25° C, based on module's input range)		±0.2%
Operational limit (0 to 60° C, based on module's input range)		±0.45%
Cable length		
- shielded	max.	200 m (660 ft.)
Supply voltage L +		none
Rated insulation voltage (+9 V against ⏚)		12 V AC
- insulation group		1×B
- tested at		500 V AC
Rated insulation voltage (inputs against +9 V)		60 V AC
- insulation group		1×B
- tested at		500 V AC
Current consumption		
- from +9 V (CPU)	typ.	70 mA
Power loss of the module	typ.	0.7 W
Weight	approx.	230 g

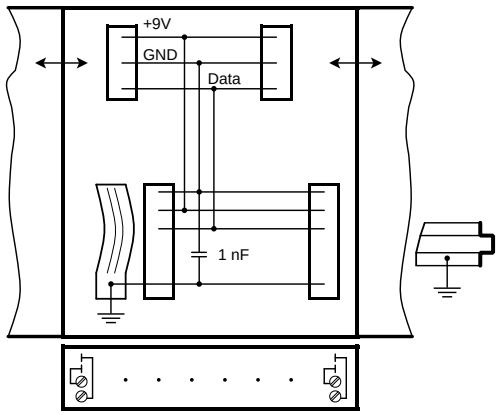
EMC Characteristics for Use in S5-95F

Electrostatic discharge to IEC 801-2 tested at:	Severity Level 3 8 kV air discharge 6 kV contact discharge cabinet mounting
Electromagnetic fields to IEC 801-3 tested at:	Severity Level 3 10 V / m
Burst to IEC 801-4 tested at:	Severity Level 4 2 kV on signal line

A.1.4 Bus Units

Bus Unit (SIGUT Screw-Type Terminals) (6ES5 700-8FA11)



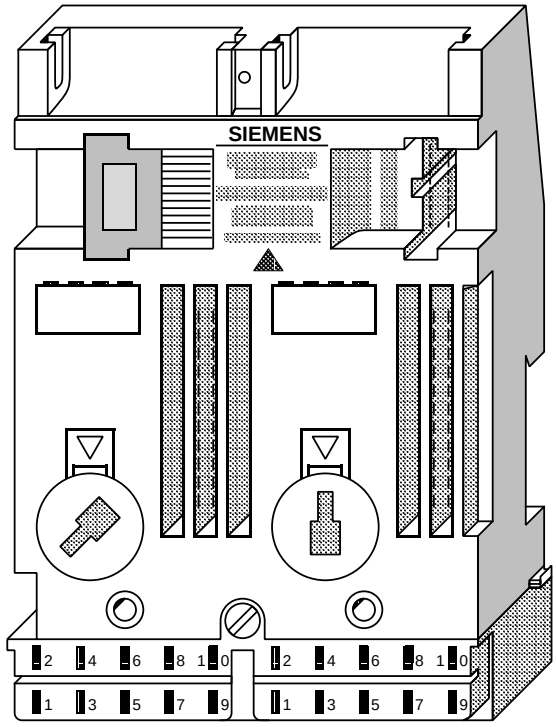


Technical Specifications

Type of connection		SIGUT screw-type terminals
Number of plug-in modules		2
Number of bus units per programmable controller	max.	16
Connection between two bus units		flat ribbon
Number of terminals		10 per slot
Insulation rating		to VDE 0160
Rated insulation voltage (+9 V to ⏚)		12 V AC
- insulation group		1×B
- tested with		500 V AC
Conductor cross section		
- stranded*		2×0.5 to 1.5 mm^2
- solid		2×0.5 to 2.5 mm^2
Current consumption		
- from +9 V (CPU)	typ.	1 mA
Dimensions		
W × H × D (in mm)		91.5×168×39
(in in.)		3.6 × 6.6 × 1.5
Weight	approx.	300 g (10.6 oz.)

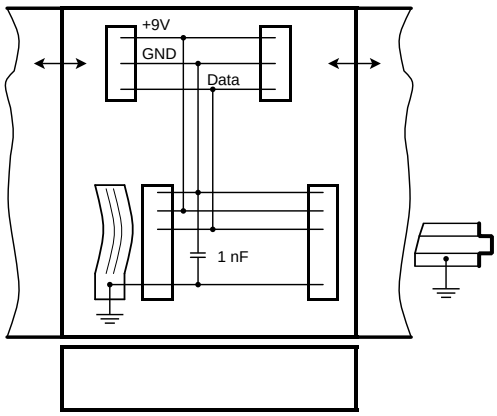
* with core end sleeves

Bus Unit (Crimp Snap-In Connections) (6ES5 700-8MA22)



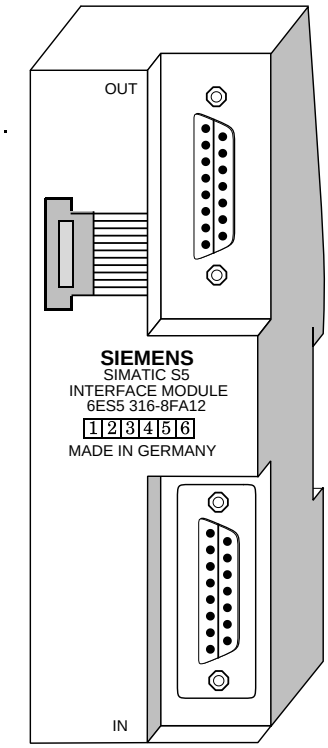
Technical Specifications

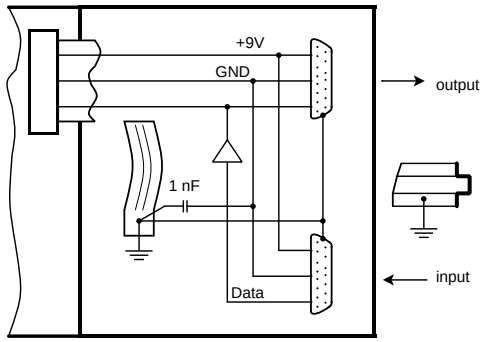
Type of connection		Crimp snap-in
Number of plug-in modules		2
Number of bus units per programmable controller	max.	16
Connection between two bus units		flat ribbon
Number of terminals		10 per slot
Conductor cross section - stranded		0.5 to 1.5 mm^2
Insulation rating		to VDE 0160
Rated insulation voltage (+9 V to ⏚)		12 V AC
- insulation group		1×B
- tested with		500 V AC
Current consumption - from +9 V (CPU)	typ.	1 mA
Dimensions W × H × D (in mm)		91.5×135×39
(in in.)		3.6× 5.3 × 1.5
Weight	approx.	250 g (8.8 oz.)



A.1.5 Interface Modules

IM 316 Interface Module (6ES5 316-8FA12)





Technical Specifications

Current supply to the expansion unit	max.	1 A
Number of interface modules per PLC	max.	4
Cable connectors for the IM 316		
- Cable connector (0.5 m/1.6 ft.)		6ES5 712-8AF00
- Cable connector (2.5 m/8.2 ft.)		6ES5 712-8BC50
- Cable connector (5.0 m/16.5 ft.)		6ES5 712-8BF00
- Cable connector (10 m/33 ft.)		6ES5 712-8CB00
Cable insulation in ducts		permissible
Permissible potential difference between ⏚(IM 316) and central ground point (CPU)		±1 V
Insulation rating		to VDE 0160
Rated insulation voltage (+9 V to ⏚)		12 V AC
- insulation group		1×B
Dimensions W × H × D (in mm)		45.4×135×39
(in in.)		1.8 × 5.3 × 1.5
Current consumption - from +9 V (CPU)	typ.	27 mA
Weight	approx.	120 g (4.2 oz.)

A.2 Standard Modules from the S5-100U Range

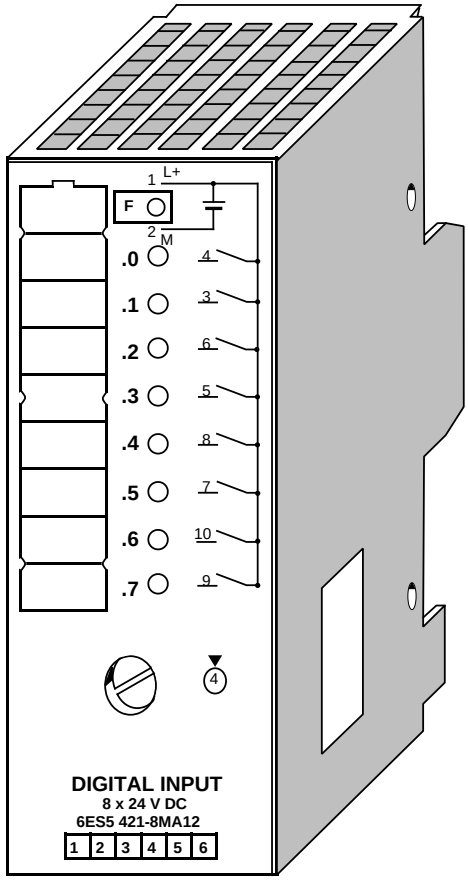
The following overview shows you the standard modules from the S5-100U system family which you can currently use in your S5-95F.

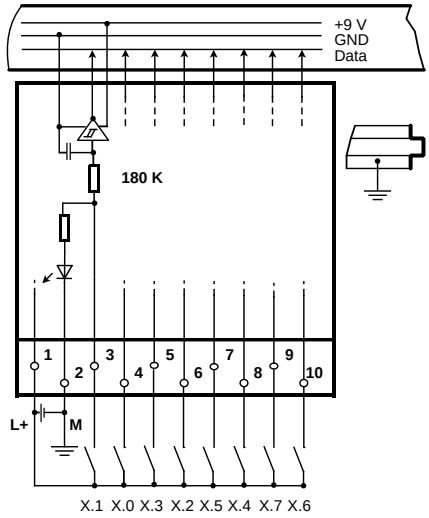
Module/Component	Order No.
Digital input module 8 x 24 V DC	6ES5 421-8MA12
Digital input module 4 x 24 to 60 V DC	6ES5 430-8MB11
Digital input module 4 x 115 V AC	6ES5 430-8MC11
Digital input module 8 x 24 V DC	6ES5 431-8MA11
Digital input module 8 x 115 V AC	6ES5 431-8MC11
Digital input module 8 x 230 V AC	6ES5 431-8MD11
Digital input module 8 x 5 to 24 V DC	6ES5 433-8MA11
Digital output module 8 x 24 V DC/0.5 A	6ES5 441-8MA11
Digital output module 4 x 24 to 60 V DC/0.5 A	6ES5 450-8MB11
Digital output module 4 x 115 to 230 V AC/1 A	6ES5 450-8MD11
Digital output module 8 x 24 V DC/1 A	6ES5 451-8MA11
Digital output module 8 x 115 to 230 V AC; 0.5 A	6ES5 451-8MD11
Relay output module 8 x 30 V DC/230 V AC	6ES5 451-8MR12
Relay output module 4 x 30 V DC/230 V AC	6ES5 452-8MR11
Digital output module 8 x 5 to 24 V DC/0.1 A	6ES5 453-8MA11
Digital input/output module with LED	6ES5 482-8MA13
Analog input module 4 x ± 50 mV	6ES5 464-8MA21
Analog input module 4 x ± 1 V	6ES5 464-8MB11
Analog input module 4 x ± 10 V	6ES5 464-8MC11
Analog input module 4 x ± 20 mA	6ES5 464-8MD11
Analog input module 4 x ± 4 to 20 mA	6ES5 464-8ME11
Analog output module 2 x ±10 V	6ES5 470-8MA12
Analog output module 2 x ± 20 mA	6ES5 470-8MB12
Analog output module 2 x ± 4 to 20 mA	6ES5 470-8MC12
Analog output module 2 x 1 to 5 V	6ES5 470-8MD12
IP 262 closed-loop control module	6ES5 262-8MA12 6ES5 262-8MB12
IP 263 positioning module	6ES5 263-8MA11
IP 264 electronic cam controller	6ES5 264-8MA11

Module/Component	Order No.
IP 266 positioning module	6ES5 266-8MA11
IP 267 stepping motor control module	6ES5 267-8MA11
Timer module 2 x 0.3 to 300 s	6ES5 380-8MA11
Counter module, 25/500 kHz	6ES5 385-8MB11
Comparator module 2 x 0.5 to 20 mA/0.5 to 10 V	6ES5 461-8MA11
CP 521 SI communications processor	6ES5 521-8MA21
CP 521 BASIC communications processor	6ES5 521-8MB12
CP 2433 master module for AS Interface	6GK1 243-3SA00
Bus unit with terminal block for screw-type connection	6ES5 700-8MA11
Bus unit with terminal block for crimp snap-in connection	6ES5 700-8MA21
IM 316 interface module	6ES5 316-8MA12

A.2.1 Digital Input Modules

Digital Input Module 8 x 24 V DC (6ES5 421-8MA12)

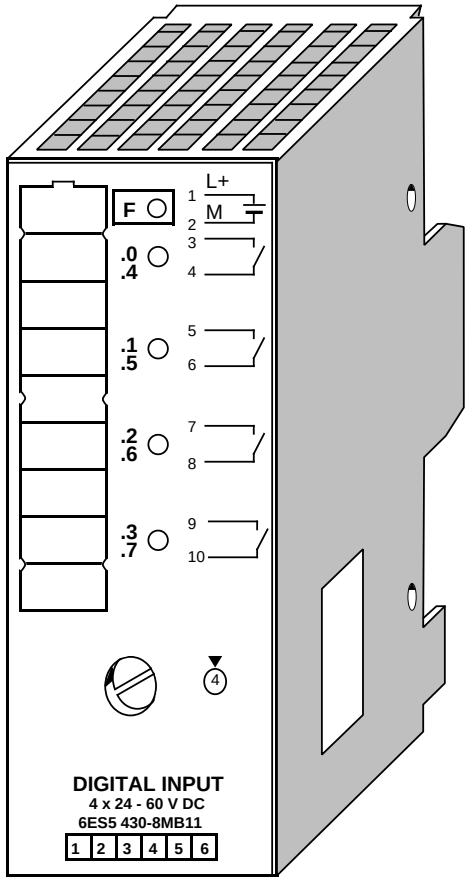


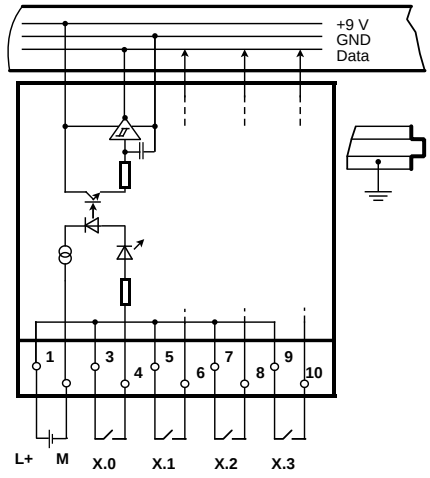


Technical specifications

Number of inputs		8
Galvanic isolation		no
- in groups of		8
Input voltage L+		
- rated value		24 V DC
- ”0” signal		0 to 5 V
- ”1” signal		13 to 33 V
Input current at ”1” signal	typ.	7 mA at 24 V
Inherent delay		
- from ”0” to ”1”	typ.	2.3 ms
- from ”1” to ”0”	typ.	4.5 ms
Length of cable		
- unshielded	max.	100 m (330 ft.)
Rated insulation voltage (+9 V to ⏚)		12 V AC
- insulation group		1×B
Fault LED (red)		no input voltage L+/M
Connection of 2-wire BERO proximity switches		possible
- residual current		1.5 mA
Current consumption from +9 V (CPU)	typ.	34 mA
Power loss of the module	typ.	1.6 W
Weight	approx.	190 g (6.7 oz.)

Digital Input Module 4 x 24 to 60 V DC (6ES5 430-8MB11)

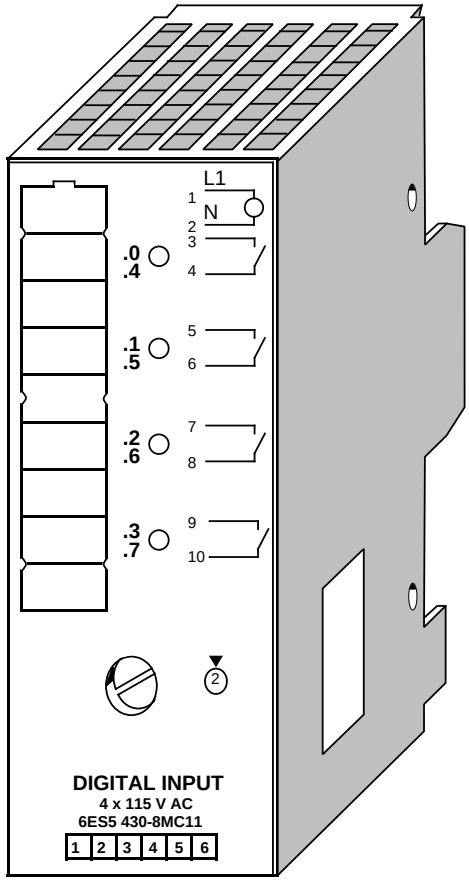


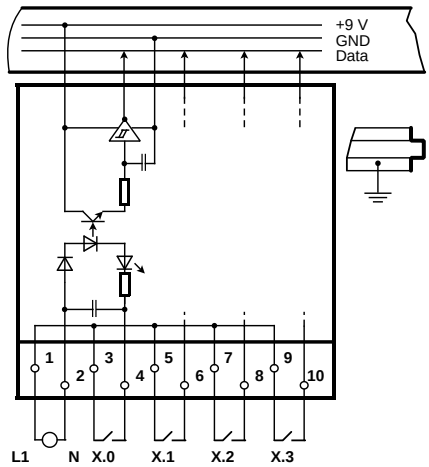


Technical specifications

Number of inputs		4
Galvanic isolation		yes (optocoupler)
- in groups of		4
Input voltage L+		
- rated value		24 to 60 V DC
- ”0” signal		- 33 to 8 V
- ”1” signal		13 to 72 V
Input current at ”1” signal	typ.	4.5 to 7.5 mA
Inherent delay		
- from ”0” to ”1”	typ.	3 ms (1.4 to 5 ms)
- from ”1” to ”0”	typ.	3 ms (1.4 to 5 ms)
Length of cable		
- unshielded	max.	100 m (330 ft.)
Rated insulation voltage (+9 V to ⏚)		12 V AC
- insulation group		1×B
- tested with		500 V AC
Rated insulation voltage (+9 V to L+)		60 V AC
- insulation group		2×B
- tested with		1250 V AC
Fault LED (red)		no input voltage L+
Connection of 2-wire BERO proximity switches		possible
- residual current		1.5 mA
Current consumption		
- from +9 V (CPU)		5 mA
- from L+	max.	35 mA
Power loss of the module	max.	2 W
Weight	approx.	200 g (7 oz.)

Digital Input Module 4 x 115 V AC (6ES5 430-8MC11)

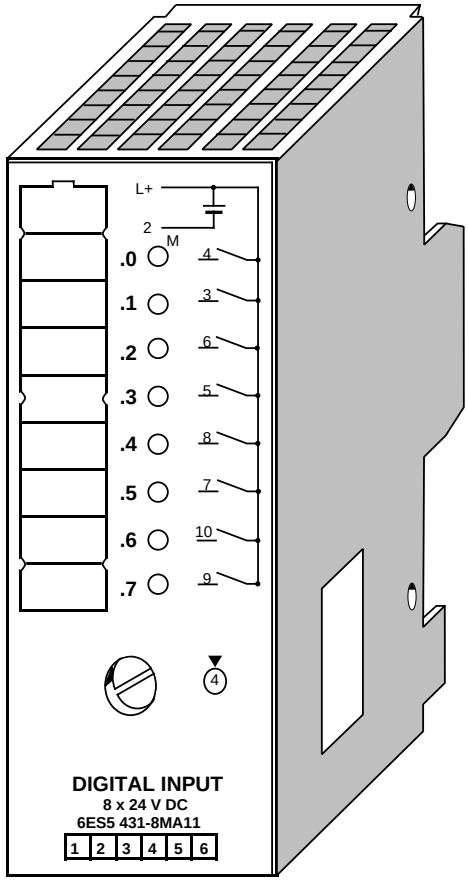


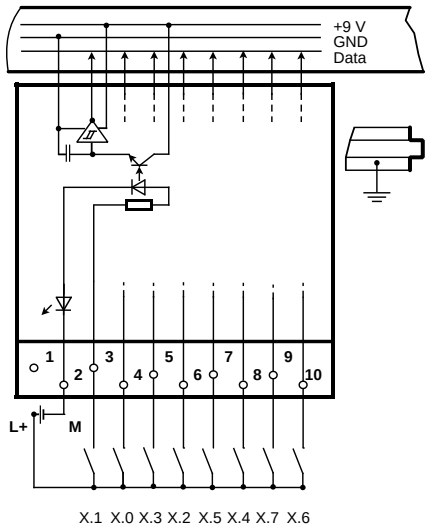


Technical specifications

Number of inputs		4
Galvanic isolation		yes (optocoupler)
- in groups of		4
Input voltage L1		
- rated value		115 V AC/DC
- ”0” signal		0 to 40 V
- ”1” signal		85 to 135 V
- frequency		47 to 63 Hz
Input current at ”1” signal	typ.	14 mA at 115 V AC
	typ.	6 mA at 115 V DC
Inherent delay		
- from ”0” to ”1”	typ.	10 ms
- from ”1” to ”0”	typ.	20 ms
Length of cable		
- unshielded	max.	100 m (330 ft.)
Rated insulation voltage (+9 V to L1)		125 V AC
- insulation group		2×B
- tested with		1250 V AC
Rated insulation voltage (+9 V to ⏚)		12 V AC
- insulation group		1×B
- tested with		500 V AC
Connection of 2-wire BERO proximity switches		possible
- residual current		5 mA
Current consumption		
- from +9 V (CPU)	typ.	16 mA
Power loss of the module	typ.	2.8 W
Weight	approx.	210 g (7.4 oz.)

Digital Input Module 8 x 24 V DC (6ES5 431-8MA11)



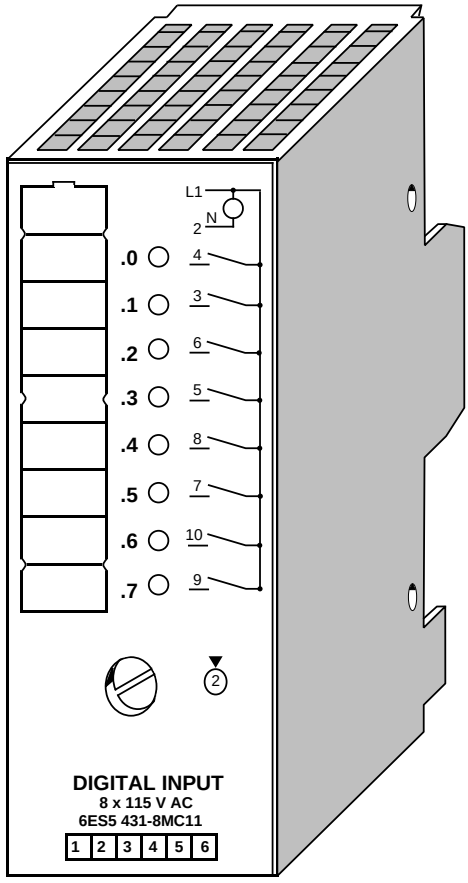


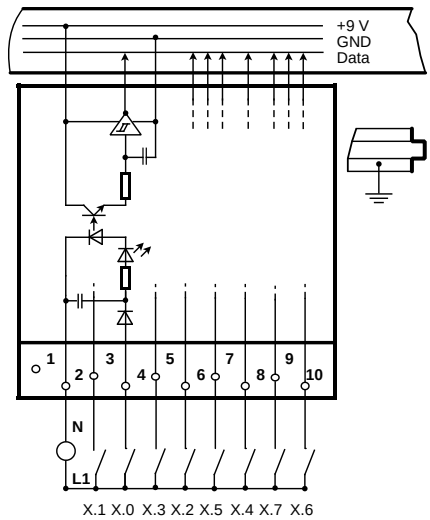
X.1 X.0 X.3 X.2 X.5 X.4 X.7 X.6

Technical Specifications

Number of inputs		8
Galvanic isolation		yes (optocoupler)
- in groups of		8
Input voltage L+		
- rated value		24 V DC
- "0" signal		0 to 5 V
- "1" signal		13 to 33 V
Input current at "1" signal	typ.	8.7 mA (at 24 V)
Inherent delay		
- from "0" to "1"	typ	5.5 ms
- from "1" to "0"	typ.	4 ms
Length of cable		
- unshielded	max.	100 m (330 ft.)
Rated insulation voltage (+ 9 V to ⏚)		12 V AC
- insulation group		2 x B
- tested with		500 V AC
Rated insulation voltage (+ 9 V to L+)		30 V AC
- insulation group		2 x B
- tested with		500 V AC
Connection of 2-wire BERO proximity switches		possible
- residual current		1.5 mA
Current consumption - from + 9 V (CPU)	typ.	32 mA
Power loss of the module	typ.	2 W
Weight	approx.	190 g (6.7 oz.)

Digital Input Module 8 x 115 V AC (6ES5 431-8MC11)

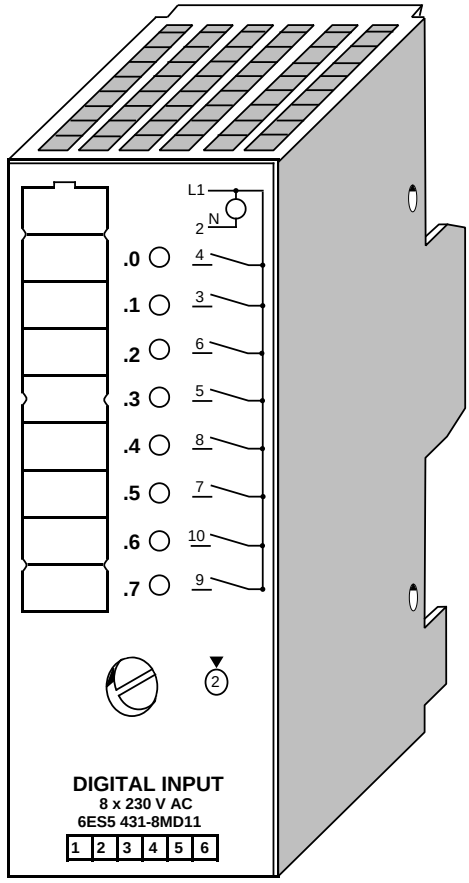


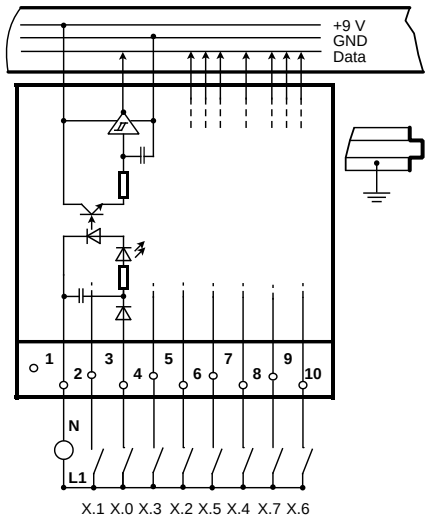


Technical specifications

Number of inputs		8
Galvanic isolation		yes (optocoupler)
- in groups of		8
Input voltage L1		
- rated value		115 V AC/DC
- "0" signal		0 to 40 V
- "1" signal		85 to 135 V
- frequency		47 to 63 Hz
Input current at "1" signal	typ.	12 mA at 115 V AC
	typ.	2.5 mA at 115 V DC
Inherent delay		
- from "0" to "1"	typ.	10 ms
- from "1" to "0"	typ.	20 ms
Length of cable		
- unshielded	max.	100 m (330 ft.)
Rated insulation voltage (+9 V to L1)		125 V AC
- insulation group		2×B
- tested with		1250 V AC
Rated insulation voltage (+9 V to ⏚)		12 V AC
- insulation group		1×B
- tested with		500 V AC
Connection of 2-wire BERO proximity switches		possible
- residual current		4 mA
Current consumption		
- from +9 V (CPU)	typ.	32 mA
Power loss of the module	typ.	2.5 W
Weight	approx.	260 g (9 oz.)

Digital Input Module 8 x 230 V AC (6ES5 431-8MD11)

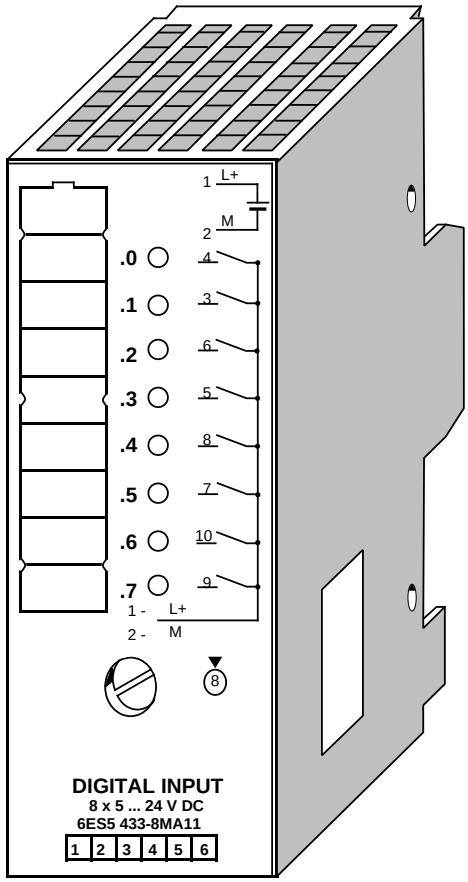


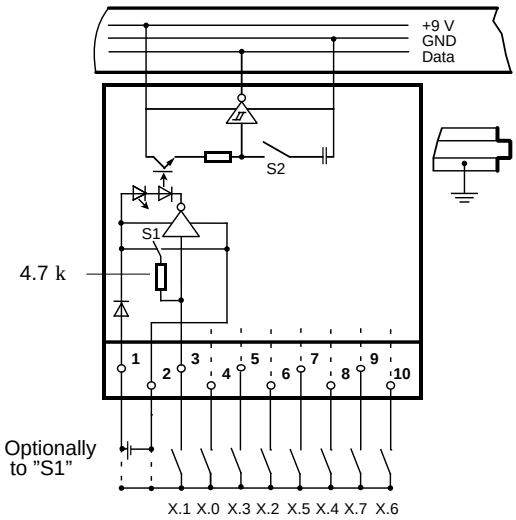


Technical specifications

Number of inputs		8
Galvanic isolation		yes (optocoupler)
- in groups of		8
Input voltage L1		
- rated value		230 V AC/DC
- "0" signal		0 to 95 V
- "1" signal		195 to 253 V
- frequency		47 to 63 Hz
Input current at "1" signal	typ.	16 mA at 230 V AC
	typ.	1.8 mA at 230 V DC
Inherent delay		
- from "0" to "1"	typ.	10 ms
- from "1" to "0"	typ.	20 ms
Length of cable		
- unshielded	max.	100 m (330 ft.)
Rated insulation voltage		
(+9 V to L1)		250 V AC
- insulation group		2×B
- tested with		1500 V AC
Rated insulation voltage		
(+9 V to ⏚)		12 V AC
- insulation group		1×B
- tested with		500 V AC
Connection of 2-wire BERO proximity switches		possible
- residual current		5 mA
Current consumption		
- from +9 V (CPU)	typ.	32 mA
Power loss of the module	typ.	3.6 W
Weight	approx.	260 g (9 oz.)

Digital Input Module 8 x 5 to 24 V DC (6ES5 433-8MA11)





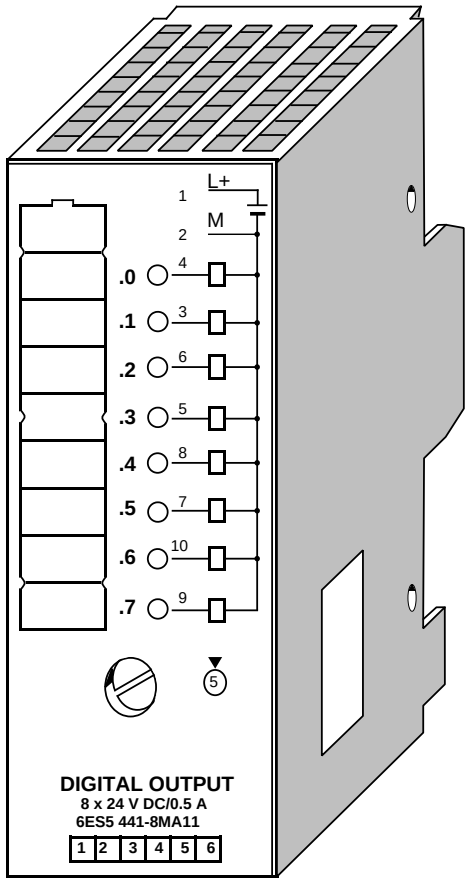
Technical Specifications

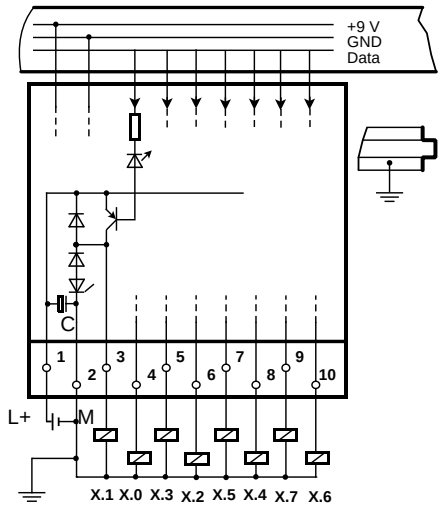
Number of inputs		8
Galvanic isolation		yes (optocoupler)
- in groups of		8
Input voltage L+		
- rated value		5 to 24 V DC
- "0" signal		Vin approx. 25% L+
- "1" signal		Vin approx. 45% L+
Permissible range		4.5 to 30 V
Input resistance		4.7 k to L+ or M; reversible on the back of the module*
The LED displays the evaluated signal		
Inherent delay	approx.	1 ms or 10 ms; reversible on the back of the module*
Length of cable		
- unshielded	max.	100 m (330 ft.)
Rated insulation voltage		
(+ 9 V to L+)		30 V AC
- insulation group		2 x B
- tested with		500 V AC
Rated insulation voltage		
(+ 9 V to ⏚)		12 V AC
- insulation group		2 x B
- tested with		500 V AC
Current consumption		
- from + 9 V (CPU)	typ.	6 mA
- from L+	typ.	60 mA
Power loss of the module	typ.	2.4 W
Weight	approx.	225 g (8 oz.)

* reversible in groups of 8

A.2.2 Digital Output Modules

Digital Output Module 8×24 V DC/0.5 A (6ES5 441-8MA11)





Technical specifications

Number of outputs		8
Galvanic isolation		no
- in groups of		8
Load voltage L+		
- rated value		24 V DC
- permissible range (including ripple)		20 to 30 V
- value at t<0.5 s		35 V

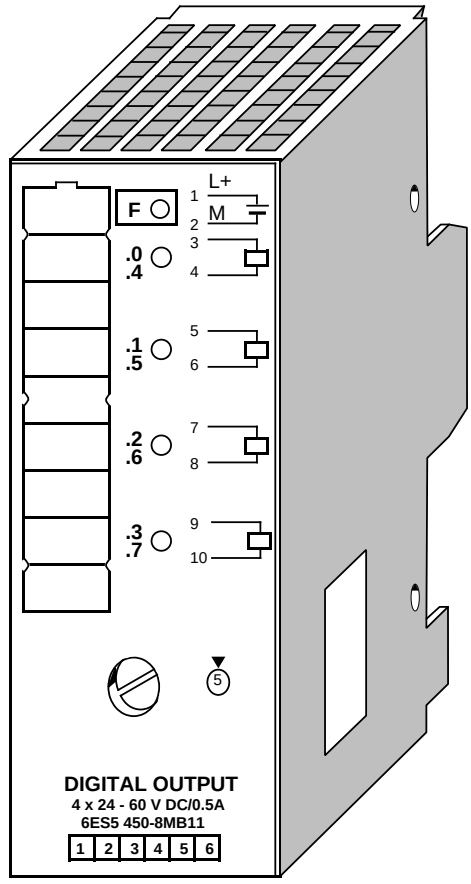
Warning

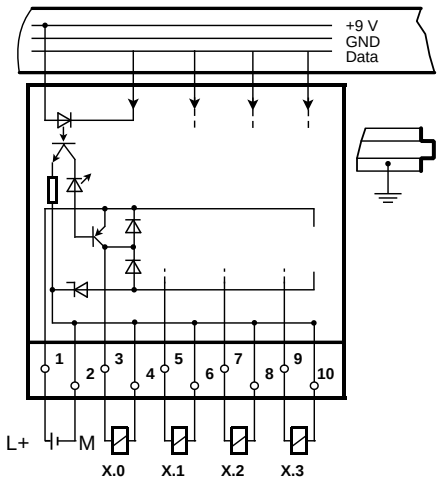
Capacitor remains loaded after switch off of L+

Output current for "1" signal		
- rated value		0.5 A at 60 °C (140 °F)/ 1 A at 30 °C (86 °F)
- permissible range		5 mA to 1 A
- lamp load	max.	5 W
Residual current at "0" signal	max.	1.0 mA
Output voltage - "1" signal	max.	L+ (- 1.2 V)
Short-circuit protection		no
Voltage induced on circuit interruption (internal) limited to		- 15 V
Switching frequency		
- resistive load	max.	100 Hz
- inductive load	max.	2 Hz
Total permissible current of outputs		4 A
Driving of digital input		possible
Paralleling of 2 outputs		possible
- maximum current		0.8 A
Length of cable		
- unshielded	max.	100 m (330 ft.)
Rated insualtion voltage* (+9 V to ⏚)		12 V AC
- insulation group		1×B
Current consumption		
- from +9 V (CPU)	typ.	14 mA
- from L+(without load)	typ.	15 mA
Power loss of the module	typ.	3.5 W
Weight	approx.	220 g (7.7 oz.)

* Relevant only for isolated assembly in the ET 100U/200 U

Digital Output Module 4×24 to 60 V DC/0.5 A (6ES5 450-8MB11)

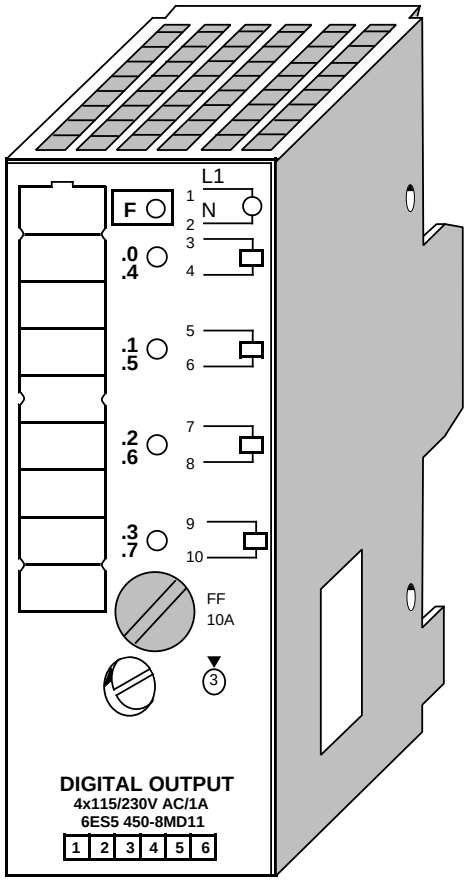


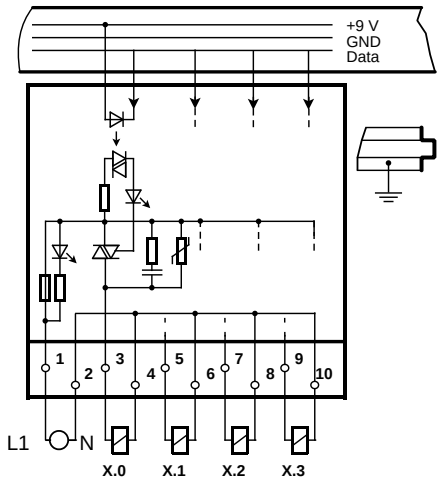


Technical specifications

Number of outputs		4
Galvanic isolation		yes (optocoupler)
- in groups of		4
Load voltage L+		
- rated value		24 to 60 V DC
- permissible range		20 to 72 V
Output current for "1" signal		
- rated value		0.5 A
- permissible range		5 mA to 0.5 A
- lamp load	max.	5 to 12 W
Residual current at "0" signal	max.	1 mA
Short-circuit protection		short-circuit protected output with autom. switch on when the short-circuit does not exist any more
Fault LED (red)		short circuit/ no load voltage L+
Error diagnostics		possible
Voltage induced on circuit interruption (internal) limited to		- 30 V
Switching frequency		
- resistive load	max.	100 Hz
- inductive load	max.	2 Hz
Total permissible current of outputs		2 A
Driving of digital input		possible
Paralleling of 2 outputs		possible
- maximum current		2×0.4 A
Length of cable		
- unshielded	max.	100 m (330 ft.)
Rated insulation voltage		
- (+9V to L+)		60 V AC
- insulation group		2×B
- tested with		500 V AC
Rated insulation voltage (+9 V to ⏚)		12 V AC
- insulation group		1×B
- tested with		500 V AC
Current consumption		
- from +9 V (CPU)	typ.	15 mA
- from L+ (without load)	typ.	30 mA (at 60 V)
Power loss of the module	typ.	5 W
Weight	approx.	200 g (7 oz.)

Digital Output Module 4×115 to 230 V AC/1 A (6ES5 450-8MD11)



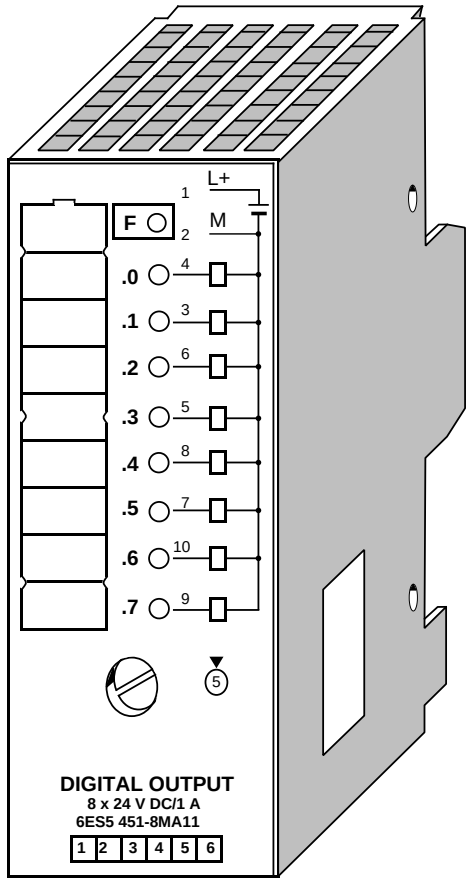


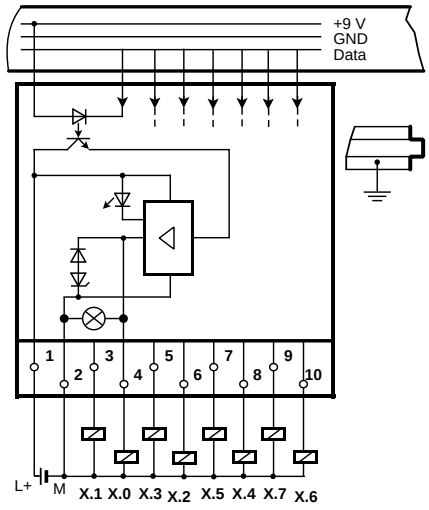
Technical specifications

Number of outputs		4
Galvanic isolation		yes
- in groups of		4
Load voltage L1		
- rated value		115 to 230 V AC
- frequency	max.	47 to 63 Hz
- permissible range		85 to 264 V
Output current for "1"signal		
- rated value		1 A
- permissible range		50 mA to 1 A
- lamp load	max.	25/50 W
Contact current closing rating:		determined by the size of the fuse
Residual current at "0" signal	max.	3/5 mA
Output voltage		
- "1" signal	max.	L1 (- 7 V)
Signal status display (green LEDs)		only with load connected
Short-circuit protection		fuse (10 A extra fast) (Wickmann No. 19231, or 6ES5 980-3BC41)
Fault LED (red)		fuse blown*
Switching frequency	max.	10 Hz
Permissible current of all outputs		4 A
Driving of digital input		possible
Paralleling of outputs		not possible
Length of cable		
- unshielded	max.	100 m (330 ft.)
Rated insulation voltage (+9 V to L1)		250 V AC
- insulation group		2×B
- tested with		1500 V AC
Rated insulation voltage (+9 V to ⏚)		12 V AC
- insulation group		1×B
- tested with		500 V AC
Current consumption		
- from +9 V (CPU)	typ.	14 mA
Power loss of the module	typ.	3.5 W
Weight	approx.	315 g (11 oz.)

* Indication only given if load voltage is applied and at least one load is connected

Digital Output Module 8 x 24 V DC/1 A (6ES5 451-8MA11)

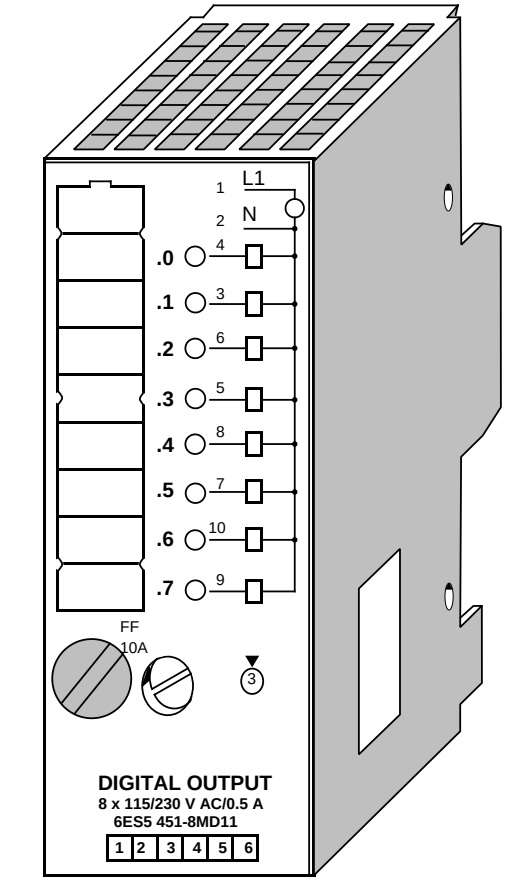


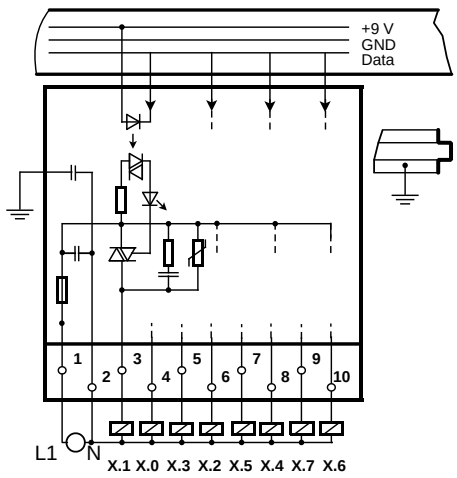


Technical specifications

Number of outputs		8
Galvanic isolation		yes (optocoupler)
- in groups of		8
Load voltage L+		
- rated value		24 V DC
- permissible range (including ripple)		20 to 30 V
- value at t<0.5 s		35 V
Output current for "1" signal		
- rated value		1 A
- permissible range		5 mA to 1A
- lamp load	max.	10 W
Residual current at "0" signal	max.	0.5 mA
Output voltage		
- at "1" signal	max.	L+ (- 0.6 V)
Short-circuit protection		short-circuit protected output with autom. switch on when the short-circuit does not exist any more
Fault LED (red)		short-circuit
Voltage induced on circuit interruption (internal) limited to		-15 V
Switching frequency		
- resistive load	max.	100 Hz
- inductive load	max.	2 Hz
Permissible current of all outputs		6 A
Driving of digital input		possible
Paralleling of 2 outputs		possible in pairs
- maximum current		1.8 A
Length of cable		
- unshielded	max.	100 m (330 ft.)
Rated insulation voltage (+ 9 V to L+)		24 V AC
- insulation group		2 x B
- tested with		500 V AC
Rated insulation voltage (+ 9 V to ⏚)		12 V AC
- insulation group		1 x B
- tested with		500 V AC
Current consumption		
- from +9 V (CPU)	typ.	35 mA
- from L+ (without load)	typ.	50 mA
Power loss of the module	typ.	3 W
Weight	approx.	230g (8 oz.)

Digital Output Module 8×115 to 230 V AC; 0.5 A (6ES5 451-8MD11)

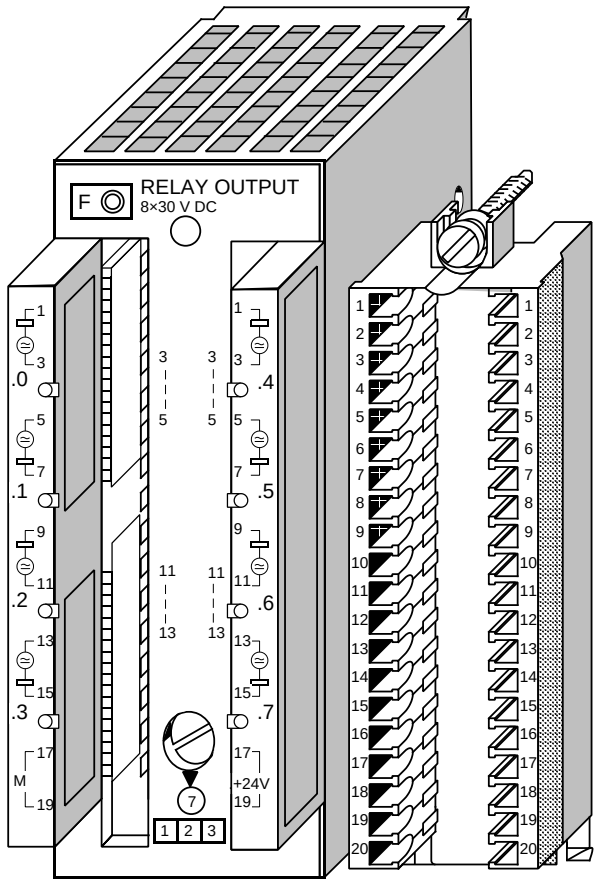


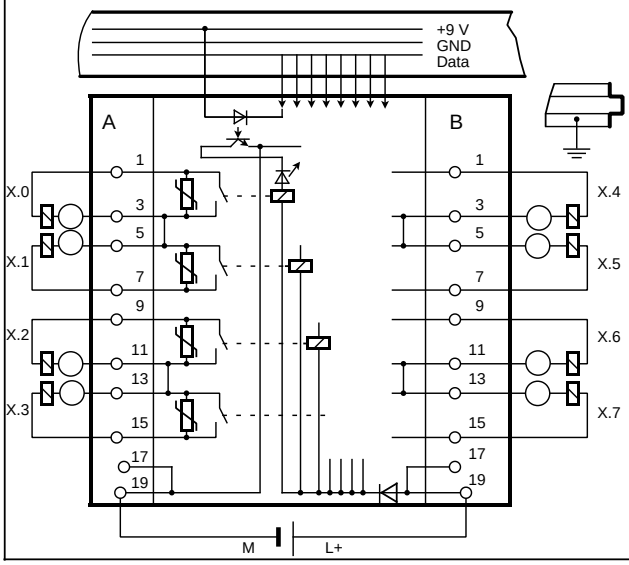


Technical specifications

Number of outputs		8
Galvanic isolation		yes (optocoupler)
- in groups of		8
Load voltage L1		
- rated value		115 to 230 V AC
- frequency	max.	47 to 63 Hz
- permissible range		85 to 264 V
Output current for "1" signal		
- rated value		0.5 A
- permissible range		50 mA to 0.5 A
- lamp load	max.	25/50 W
Contact current closing rating:		determined by the size of the fuse
Residual current at "0" signal	max.	3/5 mA
Output voltage - at "1" signal	max.	L1 (-7 V)
Signal Status Display (green LEDs)		only with load connected
Short-circuit protection		fuse (10 A extra fast) (Wickmann No. 19231, or 6ES5 980-3BC41)
Switching frequency	max.	10 Hz
Permissible current of all outputs		4 A
Driving of digital input		possible
Paralleling of outputs		not possible
Length of cable - unshielded	max.	100 m (330 ft.)
Rated insulation voltage (+9 V to L1)		250 V AC
- insulation group		2×B
- tested with		1500 V AC
Rated insulation voltage (+9 V to ⏚)		12 V AC
- insulation group		1×B
- tested with		500 V AC
Current consumption - from +9 V (CPU)	typ.	25 mA
Power loss of the module	typ.	3.5 W
Weight	approx.	270 g (9 oz.)

Relay Output Module 8 x 30 V DC/230 V AC **(6ES5 451-8MR12)**
Crimp Snap-in Connector, 40-pin **(6ES5 490-8MA12/-8MA02)**
Screw Plug Connector, 20-pin **(6ES5 490-8MB21)**
Screw Plug Connector, 40-pin **(6ES5 490-8MB11)**

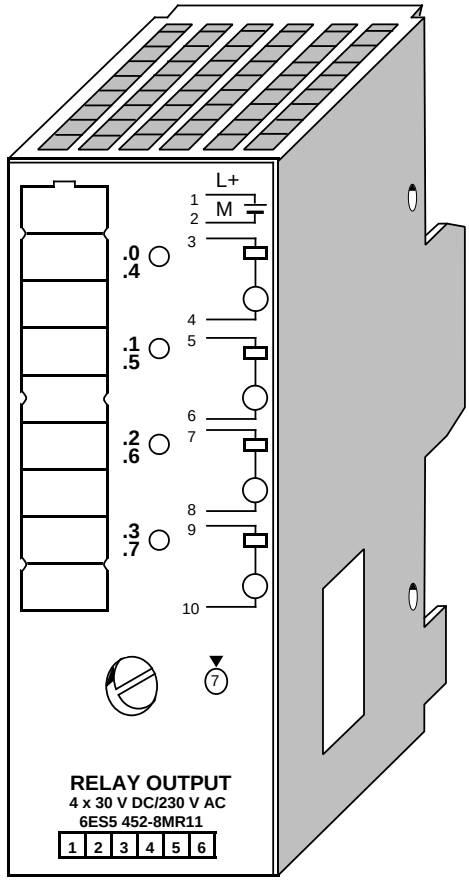


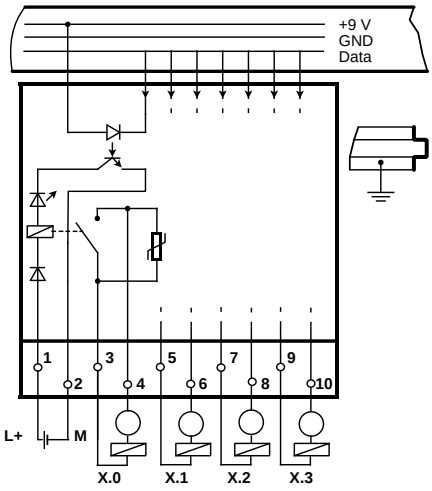


Technical specifications

Outputs		8 relay outputs, contact switching varistor
Galvanic isolation - in groups of		yes 2 with signal status display
Continuous current I_{th}		3 A
Fault LED (red)		no input voltage
Length of cable - unshielded	max.	100 m (330 ft.)
Rated insulation voltage (+ 9 V to L 1) - insulation group - tested with		250 V AC 2 x B 1500 V AC
Rated insulation voltage (+ 9 V to ⏚) - insulation group - tested with		12 V AC 1 x B 500 V AC
Current consumption - from + 9 V (CPU) - from L+	typ. typ.	30 mA 70 mA
Power loss of the module	typ.	1.6 W
Weight	approx.	300 g (11 oz.)
Switching capacity of the contacts - resistive load - inductive load	max. max.	3 A at 250 V AC 1.5 A at 30 V DC 0.5 A at 250 V AC 0.5 A at 30 V DC
Operating cycles of the contacts according to VDE 0660, part 200 - AC - 11 - DC - 11		 1×10^6 0.5×10^6
Switching frequency	max.	10 Hz
Rated insulation voltage (between contacts) - insulation group - tested with		250 V AC 2 x B 1500 V AC
Supply voltage L+ (for the relays) - rated value - ripple V_{PP} - permissible range (ripple included) - value at t <0.5 s	 max. 	 24 V DC 3.6 V 20 to 30 V 35 V

Relay Output Module 4 x 30 V DC/ 230 V AC (6ES5 452-8MR11)





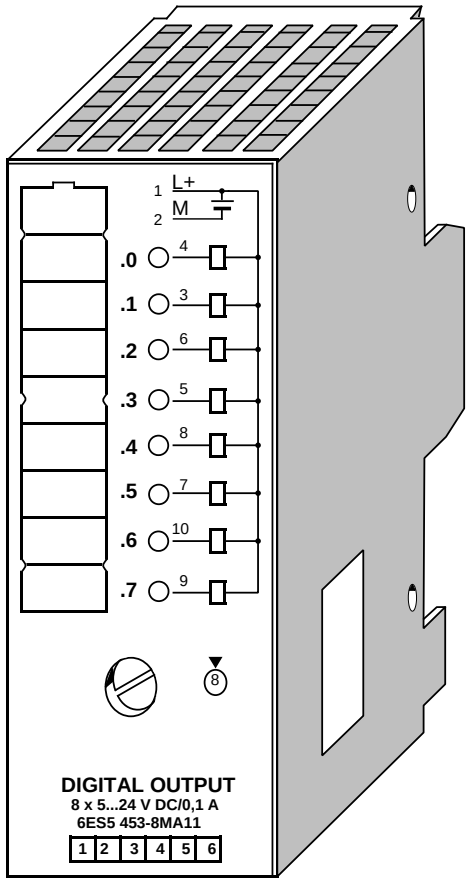
Technical specifications

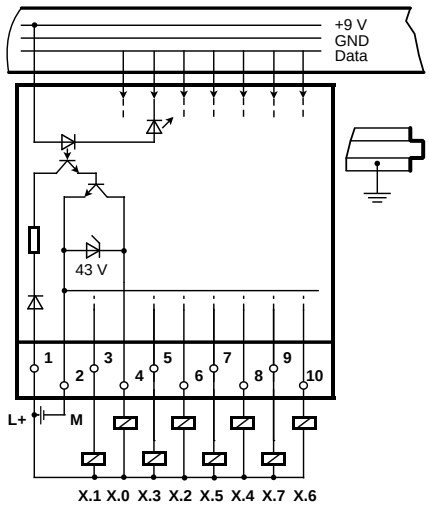
Outputs		4 relay outputs, contact switching varistor
Galvanic isolation - in groups of		yes (optocoupler) 1
Continuous current I_{th}		5 A
Length of cable - unshielded	max.	100 m (330 ft.)
Rated insulation voltage (+ 9 V to L 1) - insulation group - tested with		250 V AC 2 x B 1500 V AC
Rated insulation voltage (+ 9 V to ⏚) - insulation group - tested with		12 V AC 1 x B 500 V AC
Current consumption - from + 9 V (CPU) - from L+	typ. typ.	14 mA 100 mA
Power loss of the module	typ.	2 W
Weight	approx.	240 g (11 oz.)
Switching capacity of the contacts - resistive load - inductive load	max. max.	5 A at 250 V AC 2.5 A at 30 V DC 1.5 A at 250 V AC 0.5 A at 30 V DC
Operating cycles of the contacts according to VDE 0660, part 200 - AC - 11 - DC - 11		 1.5×10^6 0.5×10^6
Switching frequency	max.	10 Hz
Rated insulation voltage (between contacts) - insulation group - tested with		250 V AC 2 x B 1500 V AC
Supply voltage L+ (for the relays) - rated value - ripple V_{PP} - permissible range (ripple included) - value at t <0.5 s	 max. 	 24 V DC 3.6 V 20 to 30 V 35 V

Note

Relay module 452-8MR11 must never be removed or inserted while the controller is in RUN mode, as this could cause the relay outputs to be briefly activated.

Digital Output Module 8×5 to 24 V DC/0.1 A (6ES5 453-8MA11)





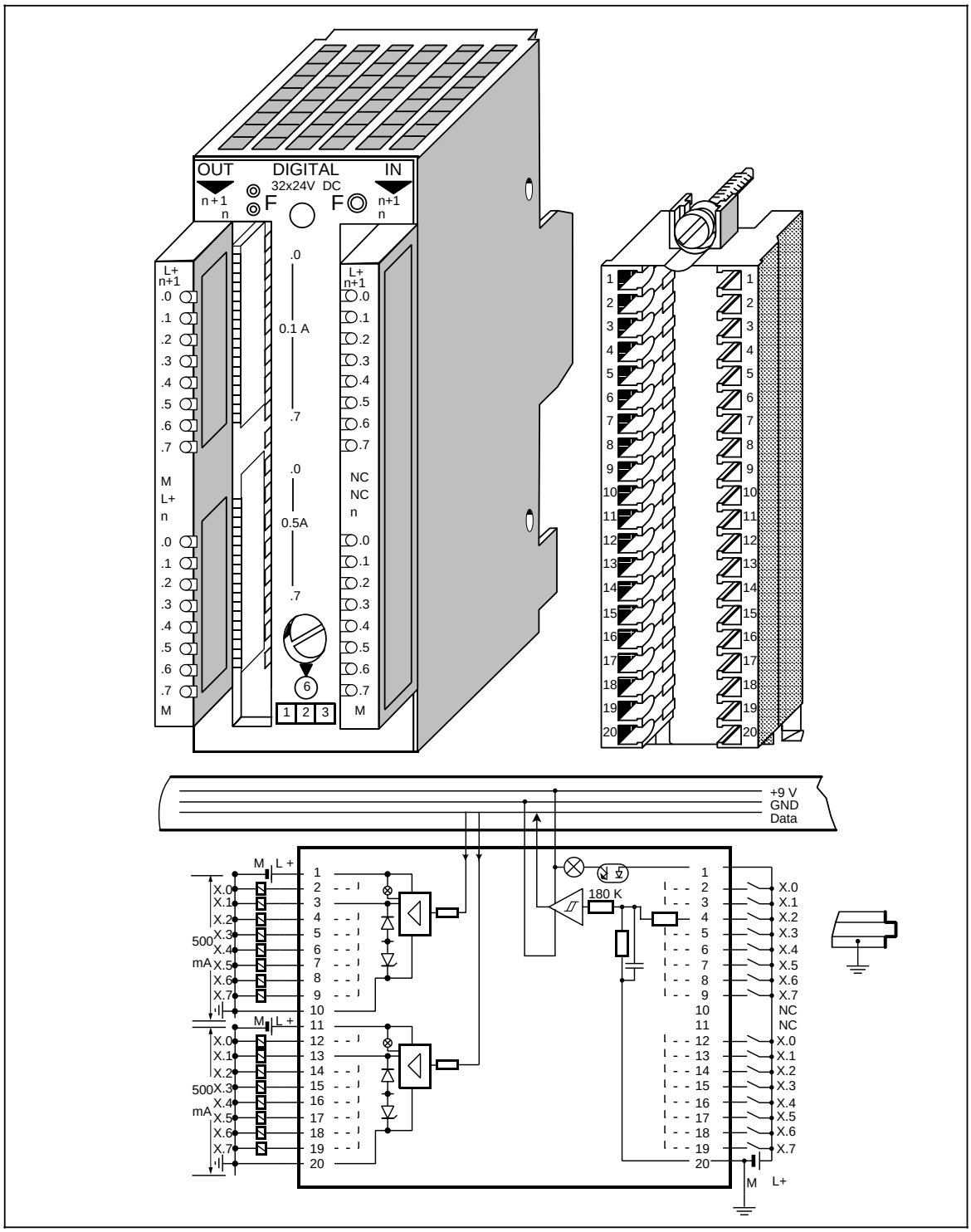
Technical specifications

Number of outputs		8
Galvanic isolation		yes
- in groups of		8
Load voltage L+		
- rated value		5 to 24 V DC
- permissible range (including ripple)		4.75 to 30 V
- value at t<0.5 s		35 V
Output current for "1" signal		
- rated value		100 mA
Output voltage		TTL-compatible*
Short-circuit protection		no
Voltage induced on circuit interruption (internal) limited to		- 19 V (at 24 V)
Switching frequency		
- resistive load	max.	100 Hz
- inductive load	max.	2 Hz
Paralleling of outputs		possible in pairs
- maximum current		(0.8 x I_{rated})
Length of cable		
- unshielded	max.	100 m (330 ft.)
Rated insulation voltage (+9 V to ⏚)		12 V AC
- insulation group		1×B
- tested with		500 V AC
Current consumption		
- from +9 V (CPU)	typ.	20 mA
- from L+ (without load)	typ.	28 mA
Power loss of the module	typ.	1 W
Weight	approx.	220 g (8 oz.)

* transistor with open collector, switching to M potential

A.2.3 Digital Input/Output Modules

Digital Input/Output Module with LED Display **(6ES5 482-8MA13)**
Crimp Snap-in Connector, 40-pin **(6ES5 490-8MA12/8MA02)**
Screw Plug Connector, 40-pin **(6ES5 490-8MB11)**



Digital Input/Output Module with LED Display (continued) (6ES5 482-8MA13)

Technical specifications

Cable length		
- unshielded		100 m (330 ft.)
Rated insulation voltage		
(+9 V to ⏚)		12 V AC
- insulation group		1 x B
Power loss of the module	typ.	4.5 W
Weight	approx.	190 g (7 oz.)

Input side

Number of inputs		16
Galvanic isolation		no
- in groups of		16
Input voltage L+		
- rated value		24 V DC
- for ”0” signal		0 to 5 V
- for”1” signal		13 to 30 V
Input current for ”1” signal	typ.	4.5 mA
Inherent delay		
- from ”0” to ”1”	typ.	4 ms
- from ”1” to ”0”	typ.	3 ms
Fault LED (red)		indicates interruption of L+/M supply
Connection of two-wire BERO proximity switches		possible
- residual current		1.5 mA
Current consumption		
- from +9 V (CPU)	typ.	50 mA

Output side

Number of outputs		16
Galvanic isolation		no
- in groups of		8
Load voltage L+		
- rated value		24 V DC
- permissible range (ripple included)		20 to 30 V
- value at t<0.5 s		35 V
Output current I_N for ”1” signal		
- rated value		500 mA
- permissible range		5 to 500 mA
Residual current for ”0” signal	max.	0.5 mA
Short-circuit protection		yes
Short-circuit indication		red LED
Output voltage for ”1” signal		L+(- 0.6 V)
Voltage induced on circuit interruption (internal) limited to		- 15 V
Switching frequency with		
- resistive load		100 Hz
- inductive load		2 Hz
Permissible total current of the outputs		6 A
Driving of a digital input		possible
Paralleling of outputs		possible in pairs
- maximum current		($0.8 \times I_N$)
Current consumption		
- from +9 V (CPU)	typ.	10 mA
- from L+ (without load)	typ.	100 mA
Lamp load	max.	5 W

A.2.4 Analog Input Modules

Analog Input Modules - Connecting Current and Voltage Sensors to Analog Input Modules

Analog input modules convert analog process signals to digital values that the CPU can process (via the process input image PII).

Observe the following rules to connect current and voltage sensors to analog input modules.

- When you have multi-channel operations, assign the channels in ascending order. This shortens the data cycle.
- Use terminals 1 and 2
  - for the connection of a compensating box (464-8MA11)
    or
  - for the supply of two-wire transducers (464-8ME11).

  Terminals 1 and 2 cannot be used with the remaining analog input modules.
- Short-circuit the terminals of unused inputs (except for 464-8ME11).
- The permissible potential difference between the common references of the inputs must not exceed 1 V. To prevent this, set the reference potentials of the sensors to a common reference potential.
- If insulated sensors are used, either the sensor or the module must be earthed.

Analog Value Representation of Analog Input Modules

Each analog process signal has to be converted into a digital format, to be stored in the process input image (PII). The analog signals are converted into a binary digit that is written in one of the following ways.

- in one byte (466-8MC11)
  or
- in two bytes (the remaining analog input modules).

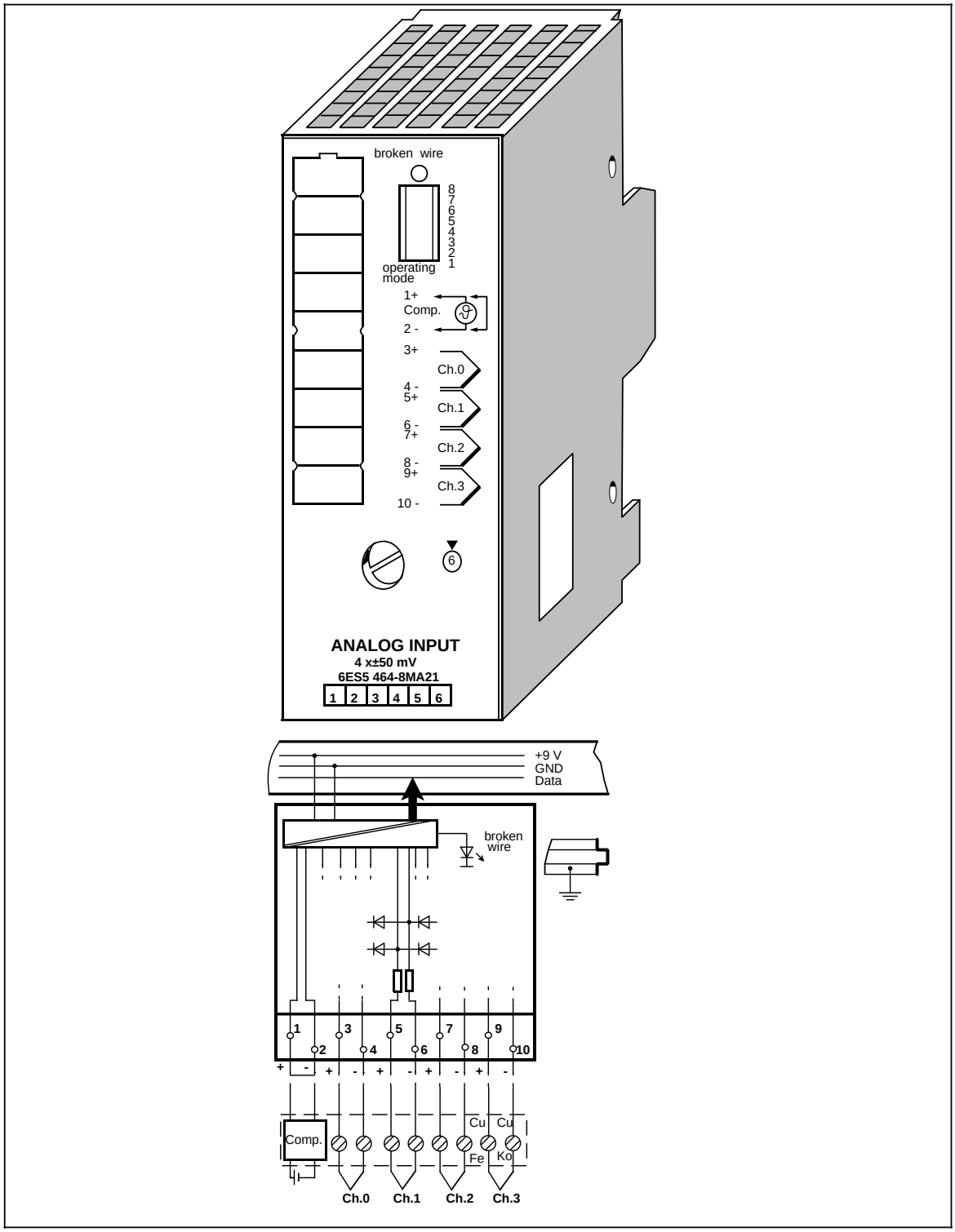
Each position has a fixed value in powers of two.
Analog values are represented in two's complement.
The following table shows the analog value representation of the different analog inputs in 2-byte format. The specific analog value representations are given in the description of the individual modules.

Tabelle A-1. Representation of an Analog Input Value as Bit Pattern

	High Byte								**Low Byte**							
Bit Number	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
Analog Value Represent.	S	2^{11}	2^{10}	2^{9}	2^{8}	2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}	X	E	OV

Key			
	S	Sign bit	0=”+”, 1=”-”
	X	Irrelevant bits	
	E	Error bit	0=no wire break; 1 = wire break
	OV	Overflow bit	0=Measured value 4095 units at the most
			1=Measured value greater than or equal to 4096 units

Analog Input Module 4 x±50 mV **(6ES5 464-8MA21)**



Analog Input Module 4 x±50 mV (continued) (6ES5 464-8MA21)

Technical specifications

Input ranges (rated values)		± 50 mV
Number of inputs		1, 2 or 4 (selectable)
Galvanic isolation		yes (inputs to grounding point; not between inputs)
Input resistance		10 M
Connection method of sensors		two-wire connection
Digital representation of input signal		12 bits + sign (2048 units = rated value)
Measured value representation		two's complement (left-justified)
Measuring principle		integrating
Conversion principle		voltage-time conversion (dual slope)
Integration time (adjustable for optimum noise suppression)		20 ms at 50 Hz 16.6 ms at 60 Hz
Encoding time per input		
- for 2048 units	max.	60 ms at 50 Hz
	max.	50 ms at 60 Hz
- for 4095 units	max.	80 ms at 50 Hz
	max.	66.6 ms at 60 Hz
Permissible voltage difference		
- between inputs	max.	±1 V
- between inputs and central ground point	max.	75 V DC/60 V AC
Permissible input voltage (destruction limit)	max.	24 V DC
Fault indication for		
- range exceeded		yes (more than 4095 units)
- sensor wire break		yes (selectable), via test pulse
- general indication of wire break		red LED

Noise suppression for f = nx (50/60 Hz±1%) n = 1, 2, ...		
- common mode rejection (V_{pp} = 1 V)		min. 86 dB
- series mode rejection (peak value of noise < rated value of input range)		min. 40 dB
Basic error limits (operating error limits at 25°C, referred to input ranges of module)		±0.15%
Operating error limits (0 to 60 °C, referred to input range of module (32 to 140 °F)		±0.4%
Linearization exactness for rated range (for types J,K,L)		±1 °C (1.8 °F)
Characteristic linearization for the following thermoelements		
- Nickel-Chromium/ Nickel-Aluminium (Type K)		IEC 584
- Iron/Copper-Nickel (Type J)		IEC 584
- Iron/Copper-Nickel (Type L)		DIN 43710
Length of cable		
- shielded	max.	50 m (164 ft.)
Supply voltage L+		none
Internal temperature compensation		possible
Connection of compensating box		possible
Rated insulation voltage (+9 V to ⏚)		12 V AC
- insulation group		1 x B
- tested with		500 V DC
Rated insulation voltage (inputs to + 9 V)		60 V AC
- insulation group		1 x B
- tested with		500 V AC
Current consumption		
- from + 9 V (CPU)	typ.	100 mA
Power loss of the module	typ.	0.7 W
Weight	approx.	230 g (8 oz.)

Function

The module 6ES5 464-8MA21 is suitable for connection of thermocouples and for voltage measurement. It features both interruptible internal ”temperature compensation” and ”linearization”.

Information on the "Design and Mode of Operation of Thermocouples" and "Connection of Thermocouples with a Compensating Box" can be found under module 6ES5 464-8MA11.

Connection Options to 6ES5 464-8MA21

There are various options for connecting thermocouples to the analog input module 6ES5 464-8MA They are obtained by combining the following criteria accordingly:

- Operation with internal/external compensation
- Operation with/without linearization
- Connection of thermocouples of the same/different type
- Direct/remote connection of the thermocouple to the module.

The table below shows the various options:

Table A-2. Options for Connecting Thermocouples

Thermocouple Characteristics		**without Linearization**		**with Linearization**	
		Direct connection to module	**Remote connection to module**	**Direct connection to module**	**Remote connection to module**
Internal Compensation	All thermocouples of the same type	not possible	not possible	possible	possible
	Thermocouples of different type	not possible	not possible	not possible	not possible
External Compensation	All thermocouples of the same type	possible	possible	not possible	not possible
	Thermocouples of different type	possible	possible	not possible	not possible

The individual options are shown below in the following connection diagrams:

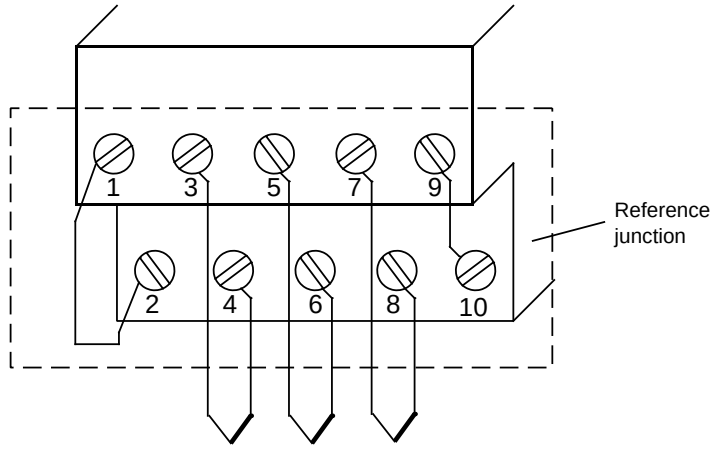


Figure A-1. Connecting Thermocouples of the Same Type with Linearization and Internal Compensation Directly to the Module

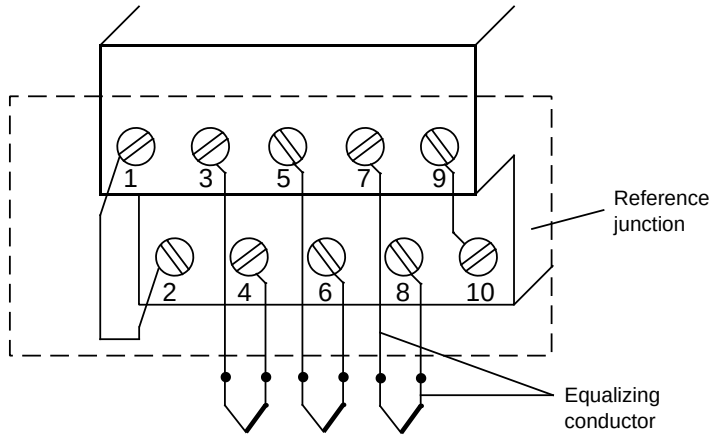


Figure A-2. Remote Connection of Thermocouples of the Same Type with Linearization and Internal Compensation to the Module

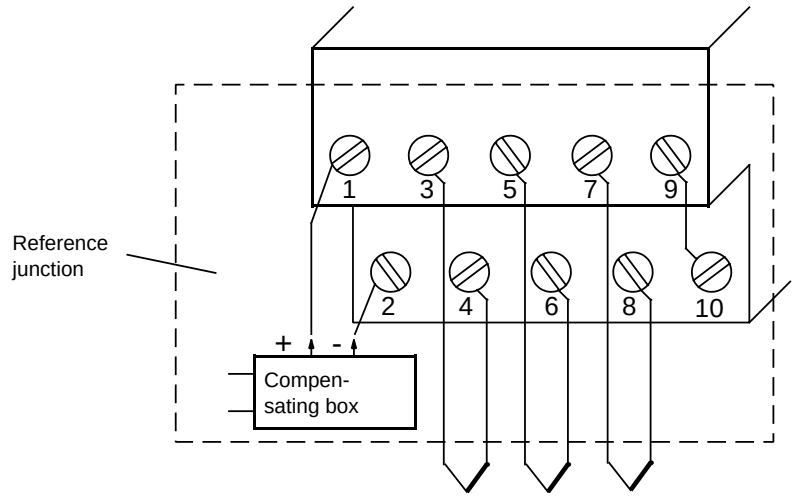


Figure A-3. Connecting Thermocouples of the Same Type without Linearization and External Compensation Directly to the Module

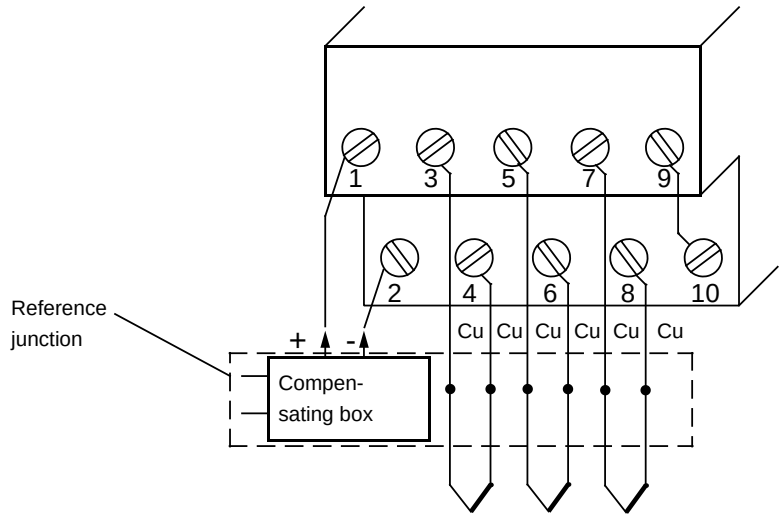


Figure A-4. Remote Connection of Thermocouples of the Same Type without Linearization and External Compensation to the Module

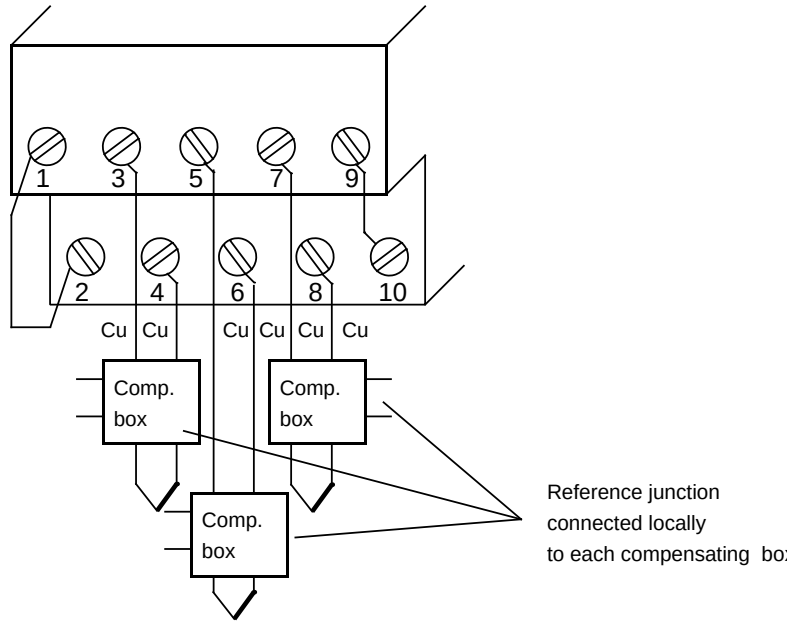


Figure A-5. Direct and Remote Connection of Thermocouples of Different Type without Linearization and External Compensation to the Module

Start-Up of the Module

Set the intended operating mode using the switch on the front panel of analog input module 464-8MA21. This switch is located on the right side at the top of the front panel of the module.

Power supply : frequency:
Set the switch to the available power supply frequency. This selects the integration time of the A/D converters for optimal interference voltage suppression.
Power frequency 50 Hz: Integration time 20 ms
Power frequency 60 Hz: Integration time 16.66 ms

Operation:
Set the number of channels you wish to assign on the analog input module. If there are fewer than four channels, the measured values will be updated faster.

Broken wire:
Once the broken wire signal has been activated, a break on one of the lines to the sensor (thermocouple) or of the sensor itself causes the red LED above the function selection switch to light up. At the same time, the broken wire error bit F (bit 1, byte 1) for the faulty channel is set.

The module "recognizes" a wire break by applying a conventional tripping current to the input terminals and by comparing the resulting voltage to a limit value. If there is a wire break in the sensor or the lines, the voltage exceeds the limit value and a "wire break" signal is sent. When the signal at the input is measured with a digital voltmeter, the tripping current pulses cause apparent fluctuations of the signal. Deactivation of the wire break signal does **not** turn off the tripping current.

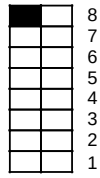
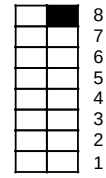
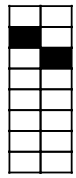
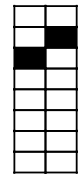
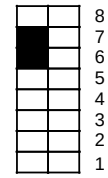
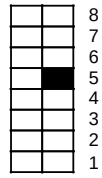
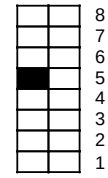
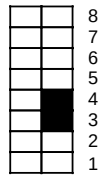
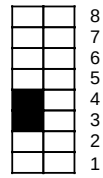
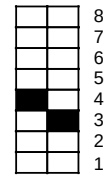
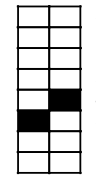
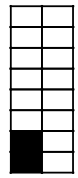
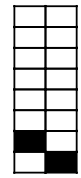
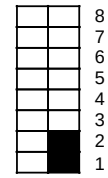
Linearization:
With this function, you can obtain a characteristic linearization of the thermocouples of type J, K, and L.
With module 464-8MA21, the linearization must always be activated together with the corresponding compensation of the reference point temperature.

Thermocouples:

Type J: - 200° C (-328° F) to	+1200° C (1392° F)	
Type K: - 200° C (-328° F) to	+1369° C (2497° F)	
Type L: - 199° C (-326° F) to	+900° C (1652° F)	(in steps each of 1° C (1.8° F)).

Temperature compensation:
For the thermocouples of type J, K, and L, you can compensate, on the one hand, the temperature of the reference point using an external compensating box.
On the other hand, it is possible to move the reference point to the front of the module by activating the "temperature compensation" function. When thermocouples are directly connected, an internal circuit on the module causes the digital value "0" to be displayed independently of the temperature of the terminal when the temperature at the measuring junction is 0° C (32° F). In order to accomplish this, the terminals of the sensors have to be connected directly or via an equalizing conductor to the module, i.e., without a copper extension cable (see Figures A.1, A.2).

Table A-3. Settings for the Operating Mode Switch for Analog Input Module 464-8MA21

Function	Settings for Operating Mode Switch			
Power supply frequency	50 Hz 	60 Hz 		
Operation	1 channel (ch. 0) 	2 channels (ch.0 u. ch1) 	4 channels (ch. 0 to 3) 	
Wire break	With wire break signal 	No wire break signal 		
Characteristic linearization of the thermocouples	No linearization 	Linearization type K 	Linearization type J 	Linearization type L 
Temperature compensation	No temperature compensation 	Temperature compensation for type K 	Temperature compensation for types J and L 	

If you have set "Characteristic linearization" and "Temperature compensation" with the operating mode switches on module 464-8MA21 for the thermocouple used, then the reference temperature is 0° C (32° F). This means that with 0° C (32° F) at the measuring junction, the value "0" is displayed.

If you equip several channels with thermocouples, use the same type of thermocouple. If you select mixed thermocouples, or if you use thermocouples other than type J, K, or L, then you must choose the following settings.

- "No linearization"
- "No temperature compensation"

When you set the switches to "no linearization" and "no temperature compensation", then module 464-8MA21 functions just like module 464-8MA11.

Analog Value Representation

The following tables show the analog value representation of the analog input module 6ES5 464-8MA21, depending on the switch position or the operating mode of the module:

Table A-4. Analog Input Module 464-8MA21, 4 x ± 50 mV, without Linearization and without Temperature Compensation (Internal) (Bipolar Fixed-Point Number)

Units	Meas. Val. in mV	High Byte								Low Byte								Range
>4095	100.0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	1	Overflow
4095	99.976	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	Overrange
2049	50,024	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	
2048	50.0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
1024	25.0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
1	0.024	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	
0	0.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Nominal range
- 1	- 0.024	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	
- 1024	- 25.0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
- 2048	- 50.0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
- 2049	- 50.024	1	0	1	1	1	1	1	1	1	1	1	1	1	0	0	0	Overrange
- 4095	- 99.976	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	
<- 4095	- 100.0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	Overflow

Standard function blocks FB 250 (read in analog value) and FB 117 (polygon function) can be used to normalize and linearize the analog value. Set type of channel y=6 ”bipolar fixed-point number” in function block FB 250.
The non-linear temperature/voltage characteristic of the thermocouples can be linearized via interpolation points using function block FB 117.
For the interpolation points please see the standard characteristic of the individual type of thermocouple in IEC 584 or DIN 43710.

Table A-5. Analog Input Module 464-8MA21, 4x±50 mV with Linearization and with Temperature Compensation (Internal); Thermocouple Type K (Nickel-Chromium/Nickel-Aluminium, according to IEC 584)

Units	Thermal Voltage in mV*	Temper-ature in °C	High Byte								Low Byte								Range
>2359			0	1	0	0	1	0	0	1	1	0	1	1	1	0	0	1	Overflow
1370		1370	0	0	1	0	1	0	1	0	1	1	0	1	0	0	0	1	Overrange**
1369	54,773	1369	0	0	1	0	1	0	1	0	1	1	0	0	1	0	0	0	
1000	41.269	1000	0	0	0	1	1	1	1	1	0	1	0	0	0	0	0	0	
500	20.640	500	0	0	0	0	1	1	1	1	1	0	1	0	0	0	0	0	
150	6.137	150	0	0	0	0	0	1	0	0	1	0	1	1	0	0	0	0	
100	4.095	100	0	0	0	0	0	0	1	1	0	0	1	0	0	0	0	0	Nominal range
1	0.039	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
- 1	- 0.039	- 1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	
- 100	- 3.553	- 100	1	1	1	1	1	1	0	0	1	1	1	0	0	0	0	0	
- 101	- 3.584	- 101	1	1	1	1	1	1	0	0	1	1	0	1	1	0	0	0	Accuracy
- 150	- 4.912	- 150	1	1	1	1	1	0	1	1	0	1	0	1	0	0	0	0	2 K
- 200	- 5.891	- 200	1	1	1	1	1	0	0	1	1	1	0	0	0	0	0	0	
- 201		- 201	1	1	1	1	1	0	0	1	1	0	1	1	1	0	0	1	Overrange**
- 273			1	1	1	1	0	0	0	0	1	0	0	1	0	0	0	1	Overflow
X		X	X	X	X	X	X	X	X	X	X	X	X	X	X	0	1	0	Wire break

* For a reference temperature of 0° C (32° F)

** In the overrange, the current slope of the characteristic curve is maintained when leaving the linearized nominal range.

Function block FB250 cannot be used.

Function block FB117 (polygon function) cannot be used, since linearization is executed by the module.

Table A-6. Analog Input Module 464-8MA21, 4x±50 mV with Linearization and with Temperature Compensation (Internal); Thermocouple Type J (Iron/Copper-Nickel (Constantan), according to IEC 584)

Units	Thermal Voltage in mV*	Tempe-rature in °C	High Byte								Low Byte								Range
1485			0	0	1	0	1	1	1	0	0	1	1	0	1	0	0	1	Overflow
1201		1201	0	0	1	0	0	1	0	1	1	0	0	0	1	0	0	1	Overrange**
1200	69.536	1200	0	0	1	0	0	1	0	1	1	0	0	0	0	0	0	0	
1000	57.942	1000	0	0	0	1	1	1	1	1	0	1	0	0	0	0	0	0	
500	27.388	500	0	0	0	0	1	1	1	1	1	0	1	0	0	0	0	0	
100	5.268	100	0	0	0	0	0	0	1	1	0	0	1	0	0	0	0	0	
1	0.05	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	Nominal range
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
- 1	- 0.05	- 1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	
- 100	- 4.632	- 100	1	1	1	1	1	1	0	0	1	1	1	0	0	0	0	0	
- 150	- 6.499	- 150	1	1	1	1	1	0	1	1	0	1	0	1	0	0	0	0	
- 199	- 7.868	- 199	1	1	1	1	1	0	0	1	1	1	0	0	1	0	0	0	
- 200	- 7.890	- 200	1	1	1	1	1	0	0	1	1	1	0	0	0	0	0	0	
- 201		- 201	1	1	1	1	1	0	0	1	1	0	1	1	1	0	0	1	Overrange**
- 273			1	1	1	1	0	1	1	1	0	1	1	1	1	0	0	1	Overflow
X		X	X	X	X	X	X	X	X	X	X	X	X	X	X	0	F	0	Wire break

* For a reference temperature of 0° C (32° F)

** In the overrange, area the current slope of the characteristic curves is maintained when leaving the linearized nominal range.

Function block FB250 cannot be used.

Function block FB117 (polygon function) cannot be used, since linearization is executed by the module.

Table A-7. Analog Input Module 464-8MA21, 4x±50 mV with Linearization and with Temperature Compensation (Internal); Thermocouple Type L (Iron/Copper-Nickel (Constantan), according to DIN 43710)

Units	Thermal Voltage in mV*	Tempe-rature in °C	High Byte								Low Byte								Range
1361			0	0	1	0	1	0	1	0	1	0	0	0	1	0	0	1	Overflow
901		901	0	0	0	1	1	1	0	0	0	0	1	0	1	0	0	1	Overrange**
900	53.14	900	0	0	0	1	1	1	0	0	0	0	1	0	0	0	0	0	
500	27.85	500	0	0	0	0	1	1	1	1	1	0	1	0	0	0	0	0	
250	13.75	250	0	0	0	0	0	1	1	1	1	1	0	1	0	0	0	0	
100	+5.37	100	0	0	0	0	0	0	1	1	0	0	1	0	0	0	0	0	Nominal range
1	0.05	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
- 1	- 0.05	- 1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	
- 100	- 4.75	- 100	1	1	1	1	1	1	0	0	1	1	1	0	0	0	0	0	
- 150	- 6.60	- 150	1	1	1	1	1	0	1	1	0	1	0	1	0	0	0	0	
- 190	- 7.86	- 190	1	1	1	1	1	0	1	0	0	0	0	1	0	0	0	0	
- 199	- 8.12	- 199	1	1	1	1	1	0	0	1	1	1	0	0	1	0	0	0	
- 200		- 200	1	1	1	1	1	0	0	1	1	1	0	0	0	0	0	1	Overrange**
- 273			1	1	1	1	0	1	1	1	0	1	1	1	1	0	0	1	Overflow
X		X	X	X	X	X	X	X	X	X	X	X	X	X	X	0	1	0	Wire break

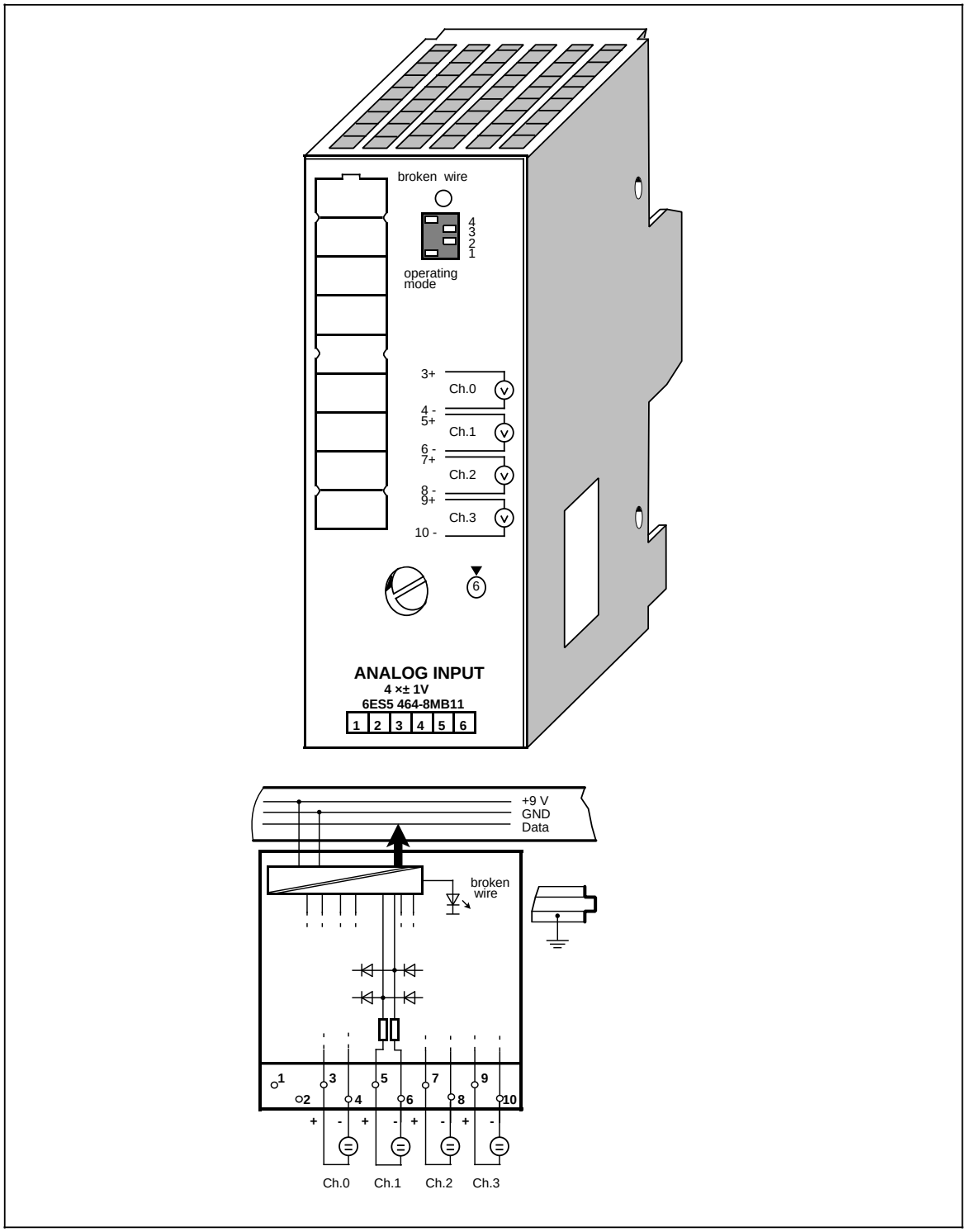
* For a reference temperature of 0° C (32° F)

** In the overrange, the current slope of the characteristic curve is maintained when leaving the linearized nominal range.

Function block FB250 cannot be used.

Function block FB117 (polygon function) cannot be used, since linearization is executed by the module.

Analog Input Module 4 x±1 V **(6ES5 464-8MB11)**



Analog Input Module 4 x ± 1 V (continued) (6ES5 464-8MB11)

Technical specifications

Input ranges (rated values)		± 1 V
Number of inputs		1, 2 or 4 (selectable)
Galvanic isolation		yes (inputs to grounding point; not between inputs)
Input resistance		10 M
Connection method of sensors		two-wire connection
Digital representation of input signal		12 bits+sign (2048 units = rated value)
Measured value representation		two's complement (left-justified)
Measuring principle		integrating
Conversion principle		voltage-time conversion (dual slope)
Integration time (adjustable for optimum noise suppression)		20 ms at 50 Hz 16.6 ms at 60 Hz
Encoding time per input		
- for 2048 units	max.	60 ms at 50 Hz
	max.	50 ms at 60 Hz
- for 4095 units	max.	80 ms at 50 Hz
	max.	66.6 ms at 60 Hz
Permissible voltage difference		
- between inputs	max.	± 1V
- between inputs and central ground point	max.	75 V DC/60 V AC
Permissible input voltage (destruction limit)	max.	24 V DC
Fault indication for		
- range exceeded		yes (more than 4095 units)
- sensor wire break		yes (selectable), via test pulse
- general indication of wire break		red LED

Noise suppression for f=nx (50/60 Hz±1%); n=1, 2, ...		
- common-mode rejection (V_{pp}=1 V)	min.	86 dB
- series-mode rejection (peak value of noise < rated value of input range)	min.	40 dB
Basic error limits (operational error limits at 25°C, referred to input ranges of the module)		±0.1 %
Operational error limits (0 to 60 °C, referred to input range of the module) (32 to 140 °F)		±0.35 %
Length of cable		
- shielded	max.	200 m (660 ft.)
Supply voltage L+		none
Rated insulation voltage (+9 V to ⏚)		12 V AC
- insulation group		1×B
- tested with		500 V AC
Rated insulation voltage (inputs to +9 V)		60 V AC
- insulation group		1×B
- tested with		500 V AC
Current consumption		
- from+9 V (CPU)	typ.	70 mA
Power loss of the module	typ.	0.7 W
Weight	approx.	230 g (8 oz.)

Function

The module 6ES5 464-8MB11 is suitable for connection of voltage sensors.

Wiring of the module (two-wire connection) is shown in the block diagram (on the first page of the module description).

Start-Up of Module

Set the intended operating mode using the switch on the front panel of analog input module 464-8MB11. The switch is located on the right side at the top of the front panel of the module.

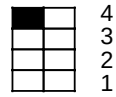
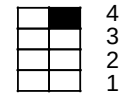
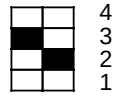
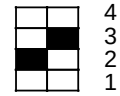
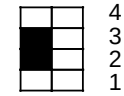
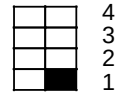
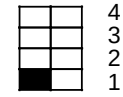
Power supply frequency: Set the switch to the available power supply frequency. This selects the integration time of the A/D converters for optimal interference voltage suppression.
Power frequency 50 Hz: Integration time 20 ms
Power frequency 60 Hz: Integration time 16.66 ms

Operation: Set the number of channels you wish to assign on the input module. If there are fewer than four channels, the measured values will be updated faster.

Broken wire: Once the broken wire signal has been activated, a break on one of the lines to the sensor or of the sensor itself causes the red LED above the function selection switch to light up. At the same time, the broken wire error bit F (bit 1, byte 1) for the faulty channel is set.

The module "recognizes" a wire break by applying a conventional tripping current to the input terminals and by comparing the resulting voltage to a limit value. If there is a wire break in the sensor or the lines, the voltage exceeds the limit value and a "wire break" signal is sent. When the signal at the input is measured with a digital voltmeter, the tripping current pulses cause apparent fluctuations of the signal. Deactivation of the wire break signal does **not** turn off the tripping current.

Table A-8. Settings for the Operating Mode Switch for Analog Input Module 464-8MB11

Function	Settings for Operating Mode Switch		
Power supply frequency	50 Hz 	60 Hz 	
Operation	1 channel (ch.0) 	2 chan. (ch.0 + ch1) 	4 channels (ch. 0 to 3) 
Wire break	With wire break signal 	No wire break signal 	

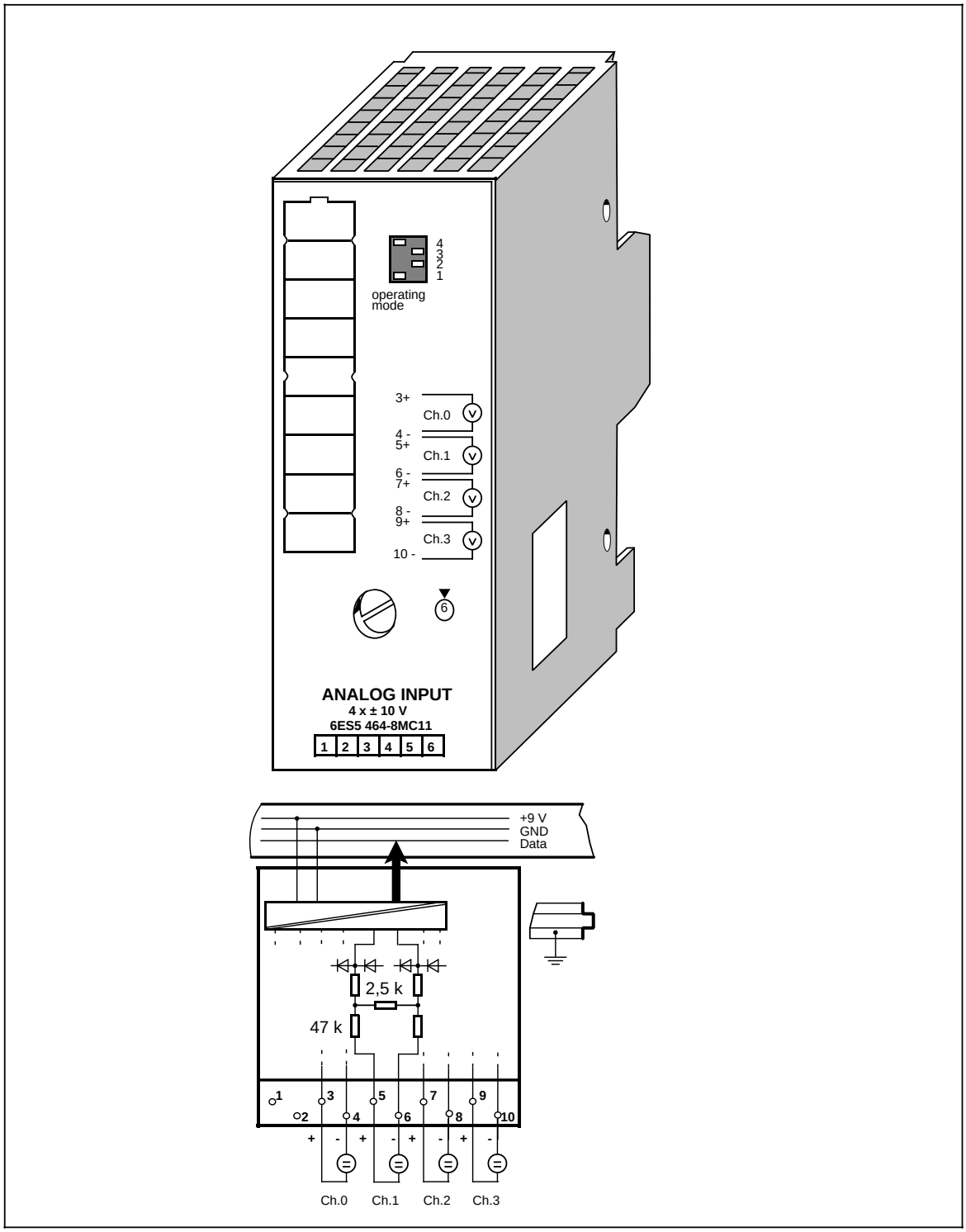
Analog Value Representation

Table A-9. Analog Input Module 464-8MB11 (Bipolar Fixed-Point Number)

Units	Meas. Val. in mV	High Byte								Low Byte								Range
>4095	2000.0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	1	Overflow
4095	1999.5	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	Overrange
2049	1000.48	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	
2048	1000.0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
1024	500.0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
1	0.48	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	
0	0.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Nominal range
- 1	- 0.48	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	
- 1024	- 500.0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
- 2048	- 1000.0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
- 2049	- 1000.48	1	0	1	1	1	1	1	1	1	1	1	1	1	0	0	0	Overrange
- 4095	- 1999.5	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	
<- 4095	- 2000.0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	Overflow

Standard function block FB250 (read in analog value) can be used to normalize the analog value. Set type of channel y=6 ”bipolar fixed-point-number” in function block FB250.

Analog Input Module 4 x±10 V **(6ES5 464-8MC11)**



Analog Input Module 4 x±10 V (continued) (6ES5 464-8MC11)

Technical specifications

Input ranges (rated values)		±10 V
Number of inputs		1, 2 or 4 (selectable)
Galvanic isolation		yes (inputs to grounding point; not between inputs)
Input resistance		50 k
Connection method of sensors		two-wire connection
Digital representation of input signal		12 bits+sign (2048 units =rated value)
Measured value representation		two's complement (left-justified)
Measuring principle		integrating
Conversion principle		voltage-time conversion (dual slope)
Integration time (adjustable for optimum noise suppression)		20 ms at 50 Hz 16.6 ms at 60 Hz
Encoding time per input		
- for 2048 units	max.	60 ms at 50 Hz
	max.	50 ms at 60 Hz
- for 4095 units	max.	80 ms at 50 Hz
	max.	66.6 ms at 60 Hz
Permissible voltage difference		
- between inputs	max.	±1 V
- between inputs and central ground point	max.	75 V DC/60 V AC
Permissible input voltage (destruction limit)	max.	50 V DC
Fault indication for		
- range exceeded		yes (more than 4095 units)
- sensor wire break		no
- general indication of wire break		no

Noise suppression for f=nx (50/60 Hz±1%); n=1,2, ...		
- common-mode rejection (V_{pp}=1 V)	min.	86 dB
- series-mode rejection (peak value of noise < rated value of input range)	min.	40 dB
Basic error limits (operational error limits at 25°C, referred to the input ranges of the module)		±0.2 %
Operational error limits (0 to 60 °C, referred to the input ranges of the module) (32 to 140 °F)		±0.45 %
Length of cable		
- shielded	max.	200 m (660 ft.)
Supply voltage L+		none
Rated insulation voltage (+9 V to ⏚)		12 V AC
- insulation group		1×B
- tested with		500 V AC
Rated insulation voltage (inputs to +9 V)		60 V AC
- insulation group		1×B
- tested with		500 V AC
Current consumption		
- from +9 V (CPU)	typ.	70 mA
Power loss of the module	typ.	0.7 W
Weight	approx.	230 g (8 oz.)

Function

The module 6ES5 464-8MC11 is suitable for connection of voltage sensors.

Wiring of the module (two-wire connection) is shown in the block diagram (on the first page of the module description).

Start-Up of Module

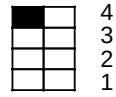
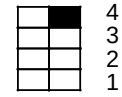
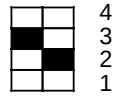
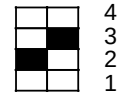
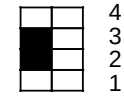
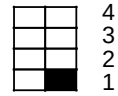
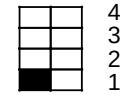
Set the intended operating mode using the switch on the front panel of analog input modules 464-8MC11. The switch is located on the right side at the top of the front panel of the module.

Power supply frequency: Set the switch to the available power supply frequency. This selects the integration time of the A/D converters for optimal interference voltage suppression.
Power frequency 50 Hz: Integration time 20 ms
Power frequency 60 Hz: Integration time 16.66 ms

Operation: Set the number of channels you wish to assign on the input module. If there are fewer than four channels, the measured values will be updated faster.

Broken wire: Broken wire signalling not possible.

Table A-10. Settings for the Operating Mode Switch for Analog Input Module 464-8MC11

Function	Settings for Operating Mode Switch		
Power supply frequency	50 Hz 	60 Hz 	
Operation	1 channel (ch.0) 	2 chan. (ch.0 + ch1) 	4 chan. (ch.0 to 3) 
No function			

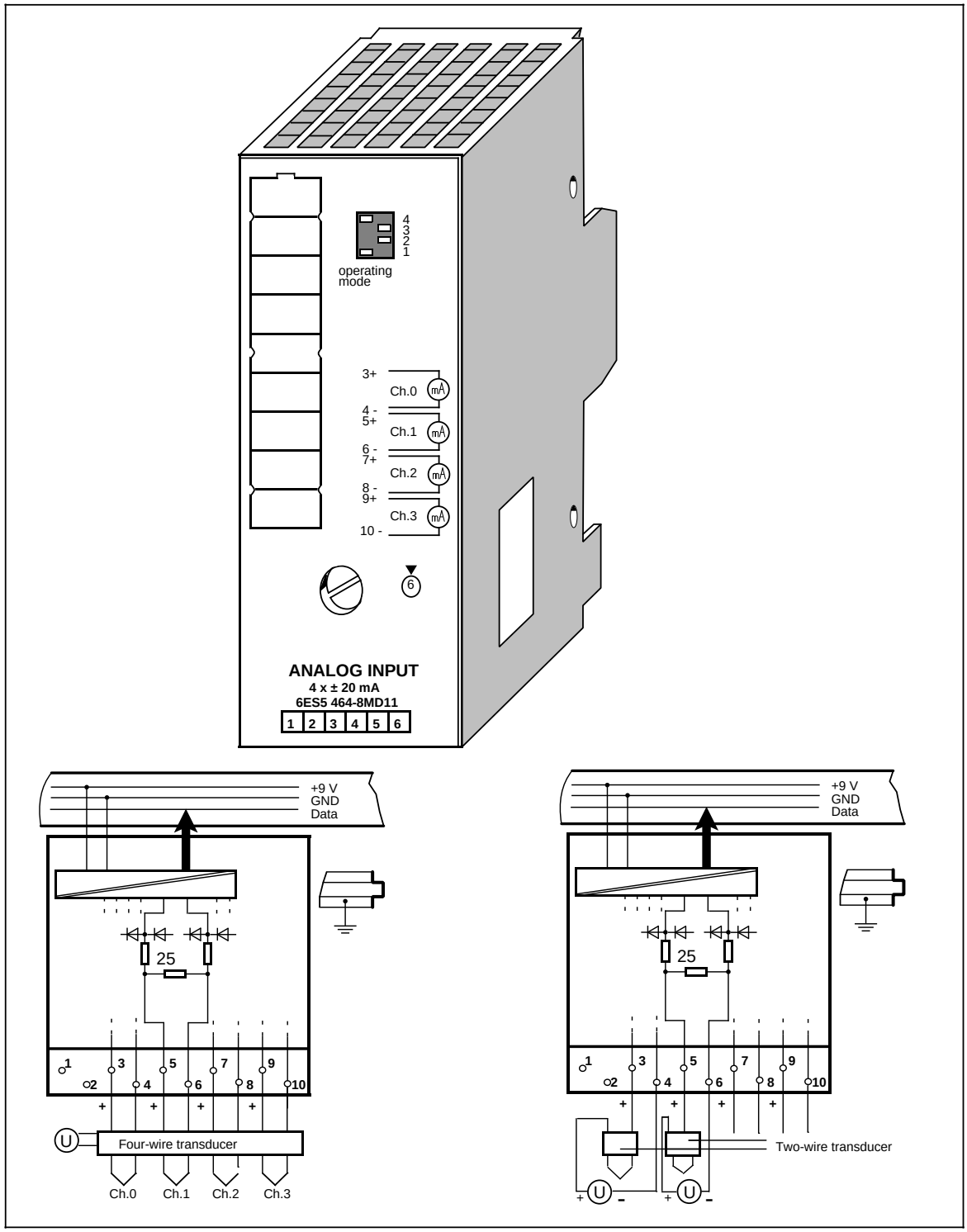
Analog Value Representation

Table A-11. Analog Input Module 464-8MC11 (Bipolar Fixed-Point Number)

Units	Meas. Val. in V	High Byte								Low Byte								Range
>4095	20.000	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	1	Overflow
4095	19.995	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	Overrange
2049	10.0048	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	
2048	10.000	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
1024	5.000	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
1	0.0048	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	
0	0.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Nominal range
- 1	- 0.0048	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	
- 1024	- 5.000	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
- 2048	- 10.000	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
- 2049	- 10.0048	1	0	1	1	1	1	1	1	1	1	1	1	1	0	0	0	Overrange
- 4095	- 19.995	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	
<- 4095	- 20.000	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	Overflow

Standard function block FB 250 (read in analog value) can be used to normalize the analog value. Set type of channel y = 6 ”bipolar fixed-point number” in function block FB250.

Analog Input Module 4 x±20 mA **(6ES5 464-8MD11)**



Analog Input Module 4 x ± 20 mA (continued) (6ES5 464-8MD11)

Technical specifications

Input ranges (rated values)		±20 V
Number of inputs		1, 2 or 4 (selectable)
Galvanic isolation		yes (inputs to grounding point; not between inputs)
Input resistance		25 k
Connection method of sensors		two-wire connection
Digital representation of input signal		12 bits+sign (2048 units =rated value)
Measured value representation		two's complement (left-justified)
Measuring principle		integrating
Conversion principle		voltage-time conversion (dual slope)
Integration time (adjustable for optimum noise suppression)		20 ms at 50 Hz 16.6 ms at 60 Hz
Encoding time per input		
- for 2048 units	max.	60 ms at 50 Hz
	max.	50 ms at 60 Hz
- for 4095 units	max.	80 ms at 50 Hz
	max.	66.6 ms at 60 Hz
Permissible voltage difference		
- between inputs	max.	±1 V
- between inputs and central ground point	max.	75 V DC/60 V AC
Permissible input voltage (destruction limit)	max.	80 mA
Fault indication for		
- range exceeded		yes (more than 4095 units)
- sensor wire break		no
- general indication of wire break		no

Noise suppression for f=nx (50/60 Hz±1%); n=1,2, ...		
- common-mode rejection (V_{pp}=1 V)	min.	86 dB
- series-mode rejection (peak value of noise < rated value of input range)	min.	40 dB
Basic error limits (operational error limits at 25°C, referred to the input ranges of the module)		±0.2 %
Operational error limits (0 to 60 °C, referred to the input ranges of the module) (32 to 140 °F)		±0.45 %
Length of cable		
- shielded	max.	200 m (660 ft.)
Supply voltage L+		none
Rated insulation voltage (+9 V to ⏚)		12 V AC
- insulation group		1×B
- tested with		500 V AC
Rated insulation voltage (inputs to +9 V)		60 V AC
- insulation group		1×B
- tested with		500 V AC
Current consumption		
- from +9 V (CPU)	typ.	70 mA
Power loss of the module	typ.	0.7 W
Weight	approx.	230 g (8 oz.)

Function

The module 6ES5 464-8MD11 is suitable for two-wire connection of current sensors.

Wiring:

If you use four-wire transducers, these must be connected as follows:

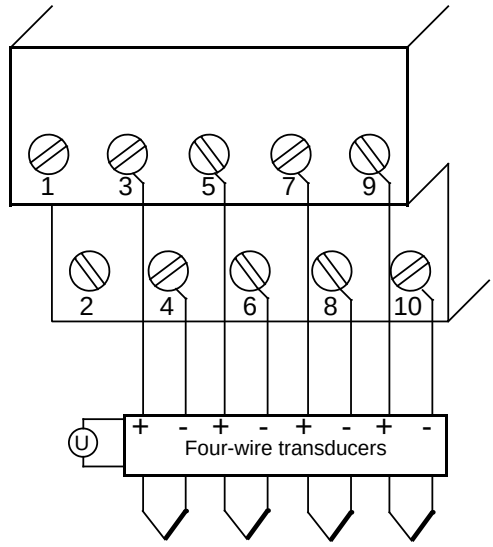


Figure A-6. Connection of only Four-Wire Transducers (6ES5 464-8MD11)

If you use two-wire transducers, these must be connected as follows:

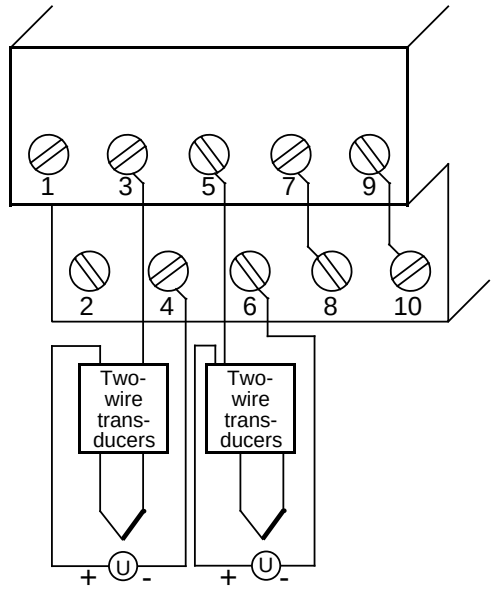


Figure A-7. Connection of only Two-Wire Transducers (6ES5 464-8MD11)

If you use two-wire and four-wire transducers, these must be connected as follows:

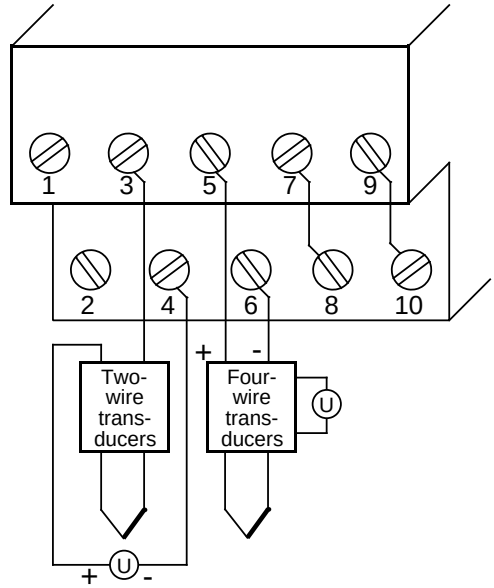


Figure A-8. Connection of Two-Wire and Four-Wire Transducers (6ES5 464-8MD11)

Start-Up of Module

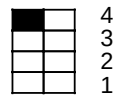
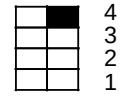
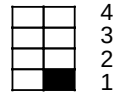
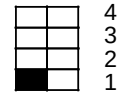
Set the intended operating mode using the switch on the front panel of analog input module 464-8MD11. This switch is located on the right side at the top of the front panel of the module.

Power supply frequency: Set the switch to the available power supply frequency. This selects the integration time of the A/D converters for optimal interference voltage suppression.
Power frequency 50 Hz: Integration time 20 ms
Power frequency 60 Hz: Integration time 16.66 ms

Operation: Set the number of channels you wish to assign on the input module. If there are fewer than four channels, less address space will be assigned and measured values will be updated faster.

Broken wire: Broken wire signalling not possible

Table A-12. Settings for the Operating Mode Switch for Analog Input Module 464-8MD11

Function	Settings for Operating Mode Switch		
Power supply frequency	50 Hz 	60 Hz 	
Operation	1 channel (ch.0) 	2 chan. (ch.0 + ch.1) 	4 chan. (ch.0 to 3) 
No function			

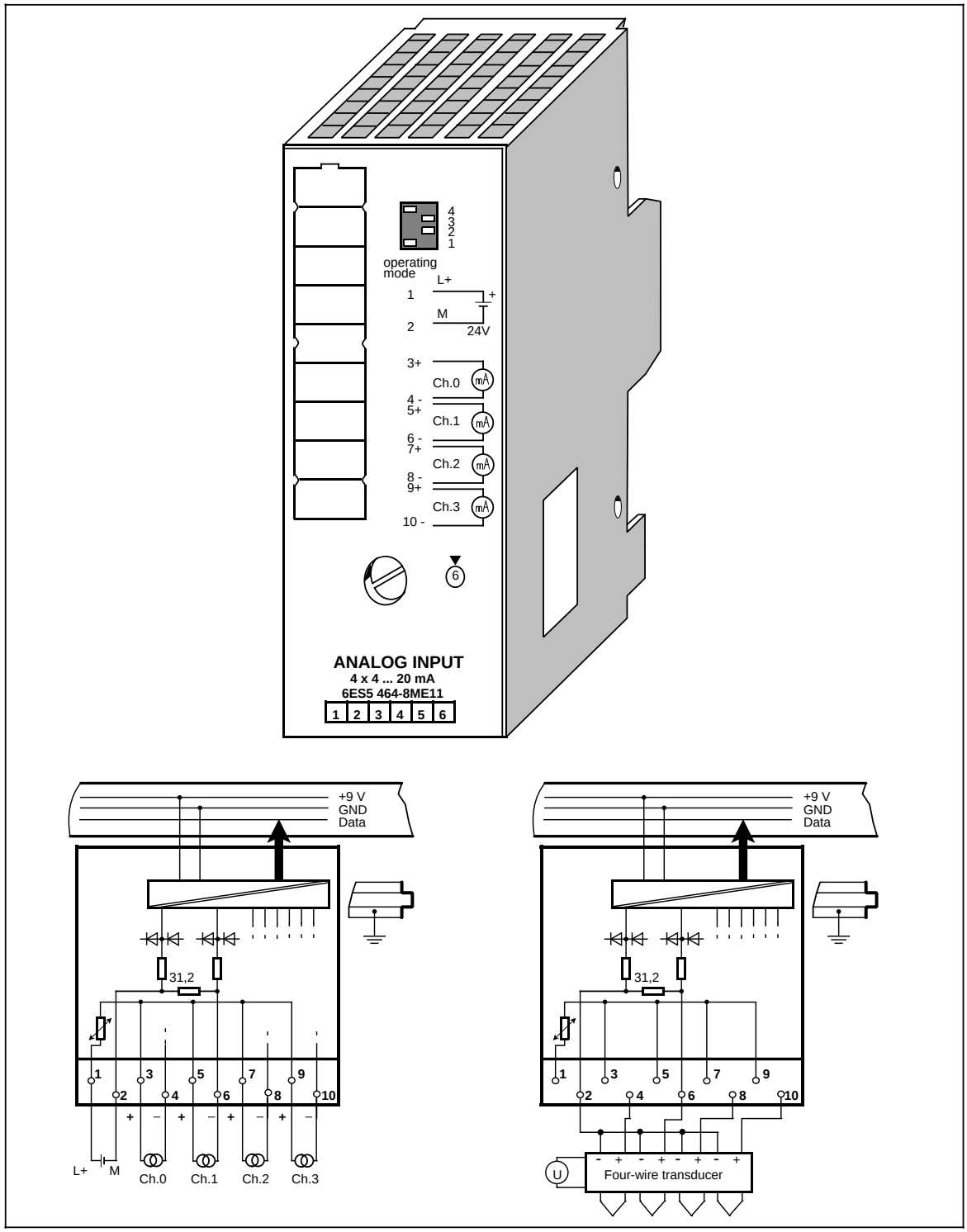
Analog Value Representation

Table A-13. Analog Input Module 464-8MD11 (Bipolar Fixed-Point Number)

Units	Meas. Val. in mA	High Byte								Low Byte								Range
>4095	40.0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	1	Overflow
4095	39.9902	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	Overrange
2049	20.0098	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	
2048	20.0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
1024	10.0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
1	0.0098	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	
0	0.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Nominal range
- 1	- 0.0098	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	
- 1024	- 10.0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
- 2048	- 20.0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
- 2049	- 20.0098	1	0	1	1	1	1	1	1	1	1	1	1	1	0	0	0	Overrange
- 4095	- 39.9902	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	
<- 4095	- 40.0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	Overflow

Standard function block FB 250 (read in analog value) can be used to normalize the analog value. Set type fo channel Y = 6 ”bipolar fixed-point number” in function block FB 250.

Analog Input Module 4 x 4 ... 20 mA **(6ES5 464-8ME11)**



Analog Input Module 4 x 4 to 20 mA (continued) (6ES5 464-8ME11)

Technical specifications

Input ranges (rated values)		4 to 20 mA
Number of inputs		1, 2 or 4 (selectable)
Galvanic isolation		yes (inputs to grounding point; not between inputs)
Input resistance		31.25 k
Connection method of sensors		two-wire connection for 2/4 wire transducers
Digital representation of input signal		12 bits+sign (2048 units =rated value)
Measured value representation		two's complement (left-justified)
Measuring principle		integrating
Conversion principle		voltage-time conversion (dual slope)
Integration time (adjustable for optimum noise suppression)		20 ms at 50 Hz 16.6 ms at 60 Hz
Encoding time per input		
- for 2048 units	max.	60 ms at 50 Hz
	max.	50 ms at 60 Hz
- for 4095 units	max.	80 ms at 50 Hz
	max.	66.6 ms at 60 Hz
Permissible voltage difference		
- between inputs	max.	±1 V
- between inputs and central ground point	max.	75 V DC/60 V AC
Permissible input voltage (destruction limit)	max.	80 mA
Fault indication for		
- range exceeded		yes (more than 4095 units)
- sensor wire break		no
- general indication of wire break		no

Noise suppression for f=nx (50/60 Hz±1%); n=1, 2, ...		
- common-mode rejection (V_{pp}=1 V)	min.	86 dB
- series-mode rejection (peak value of noise <rated value of input range)	min.	40 dB
Basic error limits (operational error limits at 25°C, referred to input ranges of the module)		±0.15 %
Operational error limits (0 to 60 °C, referred to the input ranges of the module) (32 to 140 °F)		±0.4 %
Length of cable		
- shielded	max.	200 m (660 ft.)
Supply voltage L+ for 2-wire transducers		
- rated value		24 V DC
- ripple V_{pp}		3.6 V
- permissible range		20 to 30 V
Rated insulation voltage (+9 V to ⏚)		12 V AC
- insulation group		1×B
- tested with		500 V AC
Rated insulation voltage (inputs to +9 V)		60 V AC
- insulation group		1×B
- tested with		500 V AC
Current consumption		
- from +9 V (CPU)	typ.	70 mA
- from L+	typ.	80 mA
Power loss of the module		
- for 2-wire transducers	typ.	1.0 W
- for 4-wire transducers	typ.	0.7 W
Weight	approx.	230 g (8 oz.)

Function

You can use module 6ES5 464-8ME11 for the connection of two-wire and four-wire transducers.

Use the 24 V inputs 1 and 2 to supply the two-wire transducers. The two-wire transducer converts the voltage supplied to a current of 4 to 20 mA.

Wiring:

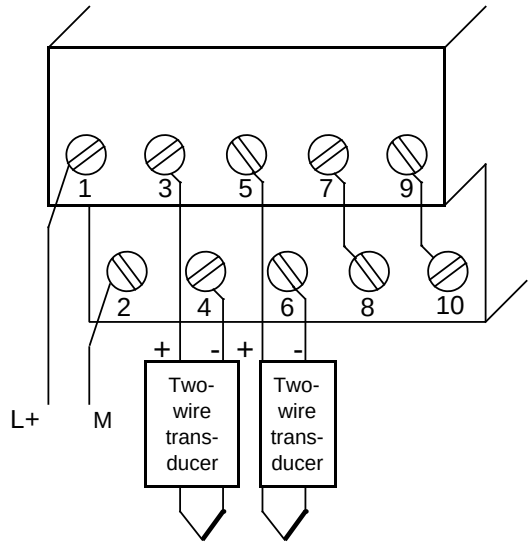


Figure A-9. Connection of only Two-Wire Transducers (6ES5 464-8ME11)

If you use a four-wire transducer connect it as shown in Figure A-10:

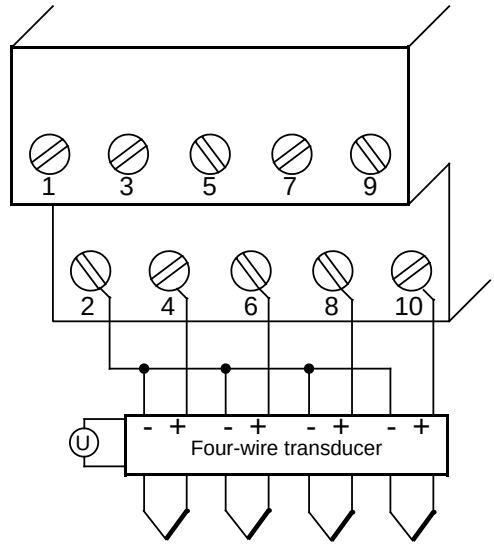


Figure A-10. Connection of only Four-Wire Transducers (6ES5 464-8ME11)

Please note that four-wire transducers require their own voltage supply and that the "+" pole of the four-wire transducer must be connected to the corresponding "-" pole of the terminal block (opposite connection technique to the two-wire transducer).
All "-" terminals of the four-wire transducer must be connected to terminal 2 of the terminal block.

The following must be observed when connecting four-wire transducers with more than two "-" terminals:

- Connect the "-" terminals to the same potential (terminal strip or terminal block) and
- Connect the terminal strip with terminal 2 of the terminal block.

Reason for this measure: Only 2 conductors may be connected at each terminal of the terminal block.

If you use two-wire and four-wire transducers simultaneously, these must be connected as follows. Please note that the four-wire transducers must be galvanically isolated from the module via isolating transformers.

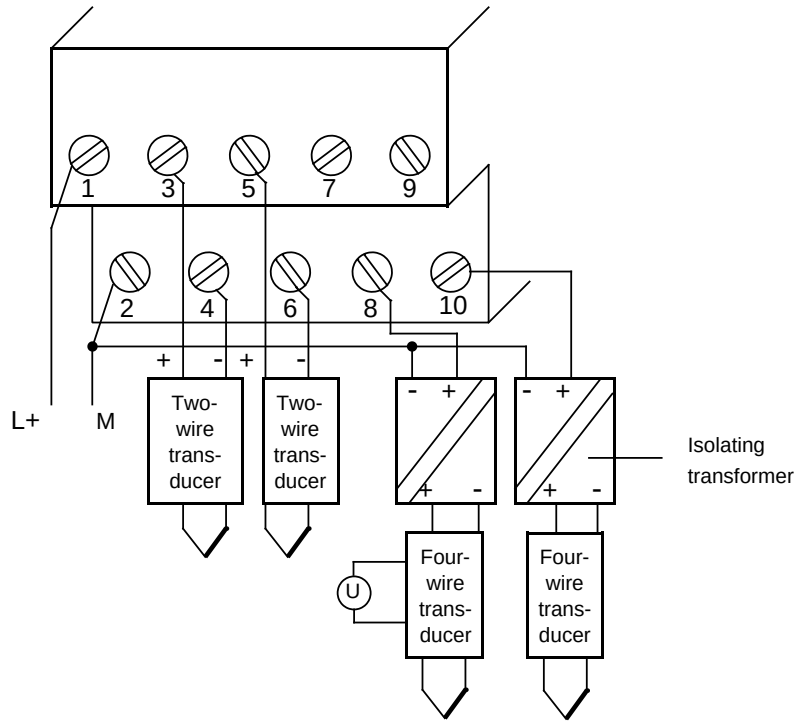


Figure A-11. Connection of Two-Wire and Four-Wire Transducers (6ES5 464-8ME11)

Start-Up of Module

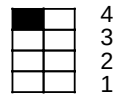
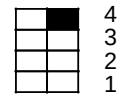
Set the intended operating mode using the switch on the front panel of analog input module 464-8ME11. This switch is located on the right side at the top of the front panel of the module.

Power supply frequency: Set the switch to the available power supply frequency. This selects the integration time of the A/D converters for optimal interference voltage suppression.
Power frequency 50 Hz: Integration time 20 ms
Power frequency 60 Hz: Integration time 16.66 ms

Operation: Set the number of channels you wish to assign on the input module. If there are fewer than four channels, the measured values will be updated faster.

Broken wire: Inputs 4, 6, 8 and 10 are connected internally via shunt resistors.
Because of the internal shunt resistors, broken wire signalling is not possible.

Table A-14. Settings of the Operating Mode Switch for Analog Input Module 464-8ME11

Function	Settings for Operating Mode Switch		
Power supply frequency	50 Hz 	60 Hz 	
Operation	1 channel (ch.0) 	2 chan. (ch.0 + ch1) 	4 chan. (ch.0 to 3) 
No function			

Analog Value Representation

Table A-15. Analog Input Module 464-8ME11, 4 x 4 to 20 mA (Absolute Value)

Units	Meas- Val. in mA	High Byte								Low Byte								Range
>4095	>32.769	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	1	Overflow
4095	31.992	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	Overrange
2561	20.008	0	1	0	1	0	0	0	0	0	0	0	0	1	0	0	0	
2560	20.0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	
2048	16.0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Nominal range
512	4.0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	
511	3.992	0	0	0	0	1	1	1	1	1	1	1	1	1	0	0	0	
384	3.0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	
0	0.0*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Transducer
- 1	- 0.008	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	failure
<- 4095	<- 32.769	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	

* Because of tolerances of components used in the module, the converted value can also be negative (e.g. $FFF8_H$ Unit: - 1).

Standard function block FB 250 (read in analog value) can be used to normalize the analog value. Set type of channel y = 3 ”absolute value” in function block FB250.

A.2.5 Analog Output Modules

Connection of Loads to Analog Output Modules

Analog output modules convert the bit patterns that are output by the CPU into analog output voltages or currents.

No adjustments are necessary if you want to connect loads to the analog outputs.

Check the following items before connecting loads.
- The load voltage 24 V DC must be connected to terminals 1 and 2.
- The maximum permissible potential difference between the outputs is 60 V AC (for this purpose it might be necessary to earth the actuator or the module).
- Unused outputs must be left open-circuited.

Analog Value Representation of the Analog Output Modules

The following table shows how the analog value to be output is stored in the PIQ. The special analog value representations are included in the descriptions of the individual modules.

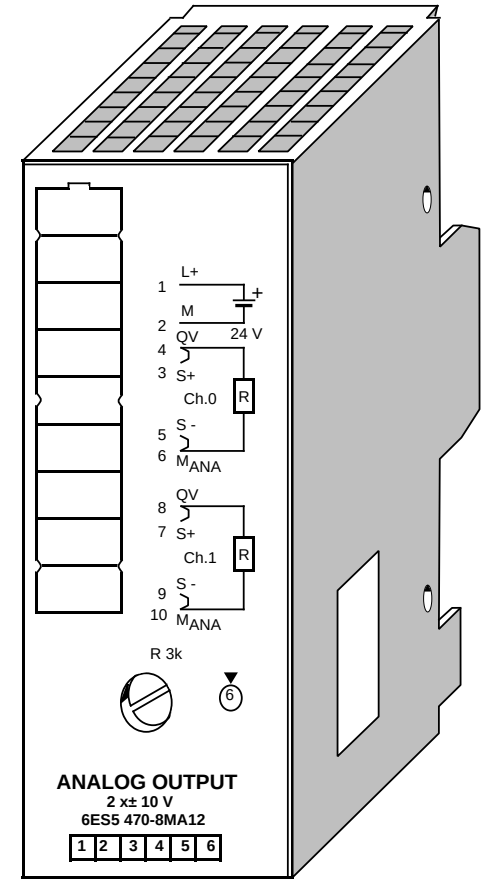
Table A-16. Representation of an Analog Output Value as Bit Pattern

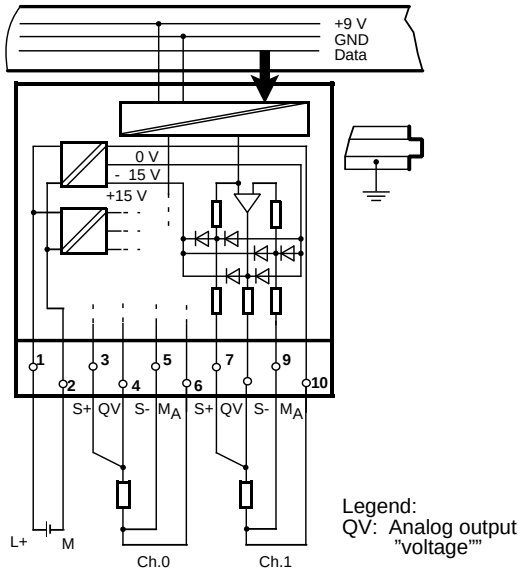
	High Byte								Low Byte							
Bit Number	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
Rep. of Analog Value	VZ	2^{10}	2^{9}	2^{8}	2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}	X	X	X	X

Key: X irrelevant bits

Standard function block FB 251 (output analog value) can be used for all analog output modules 470-8M...12 to support the analog value output.

Analog Output Module 2 x±10 V (6ES5 470-8MA12)





Legend:
QV: Analog output "voltage""

Technical specifications

Output range (rated values)		±10 V
Number of outputs		2
Galvanic isolation		yes (outputs to grounding point and between outputs)
Input resistance		3.3 k
Capacitive load incl. cable capacitance		<100 nF
Connection method		two- or four-wire connection
Digital representation of output signal		11 bits + sign (1024 units = rated value)
Measured value representation		two's complement (left-justified)
Conversion time (0 to 100%)	max.	0.15 ms
Permissible overload		25%
Short-circuit protection		yes
Short-circuit current		±30 mA
Permissible voltage difference to ground and between outputs	max.	75 V DC/60 V AC
Basic error limits (operational error limits at 25°C, referred to output range of the module)		±0.3%
Operational error limits (0 to 60 °C, referred to output range) (32 to 140 °F)		±0.6%
Length of cable - shielded	max.	200 m (660 ft.)
Supply voltage L+ (peripheral)		
- rated value		24 V DC
- ripple V_{PP}		3.6 V
- permissible range (ripple included)		20 to 30 V
Rated insulation voltage (+9 V to ⏚)		12 V AC
- insulation group		1 x B
- tested with		500 V AC
Rated insulation voltage (Output to L +, between outputs, output to + 9V)		60 V AC
- insulation group		1 x B
- tested with		500 V AC
Current consumption		
- from + 9 V (CPU)	typ.	170 mA
- from L +	typ.	100 mA
Power loss of the module	typ.	3.1 W
Weight	approx.	290 g (10 oz.)

Connection of the Module

Figure A-12 shows how to connect loads to the voltage outputs of the modules.

The sensor lines (S+ and S-) must be directly connected to the load, so that the voltage is measured and regulated directly at the load. In this manner, voltage drops of up to 3 V per line can be compensated for.
The sensor lines can be left out if the resistances of the QV and M lines are negligible compared to the load resistance. In such a case, connect terminal S + to terminal QV, and terminal S to M_{ANA}.

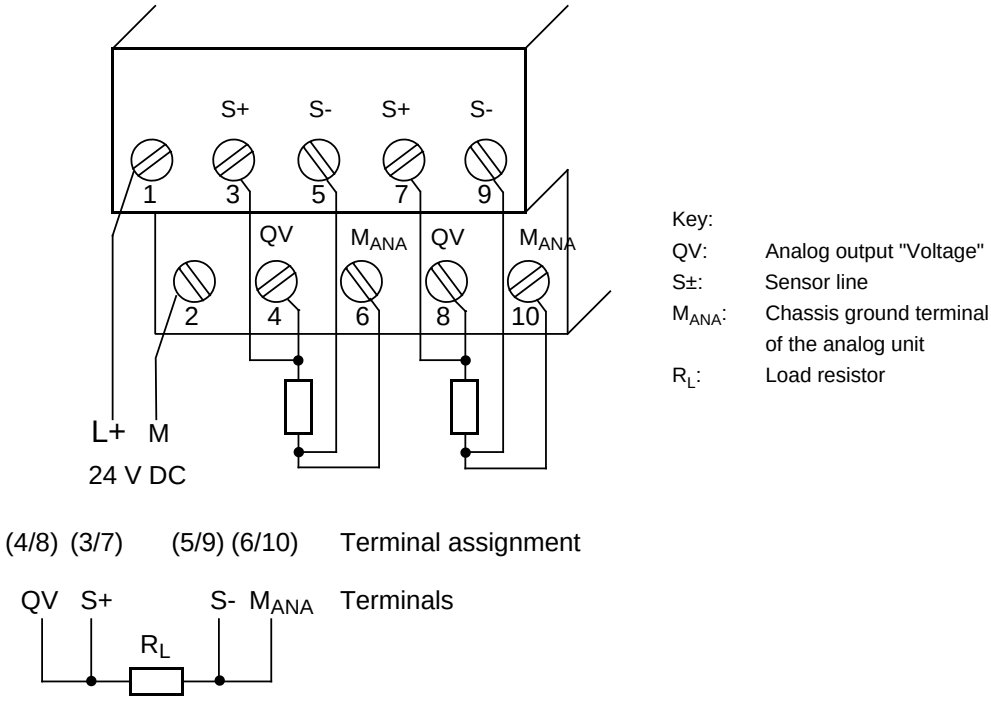


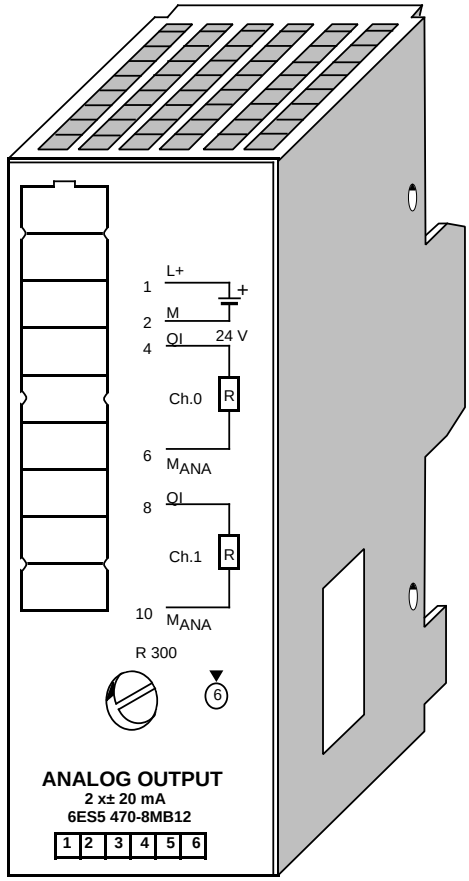
Figure A-12. Load Connection via a Four-Wire Circuit (6ES5 470-8MA12)

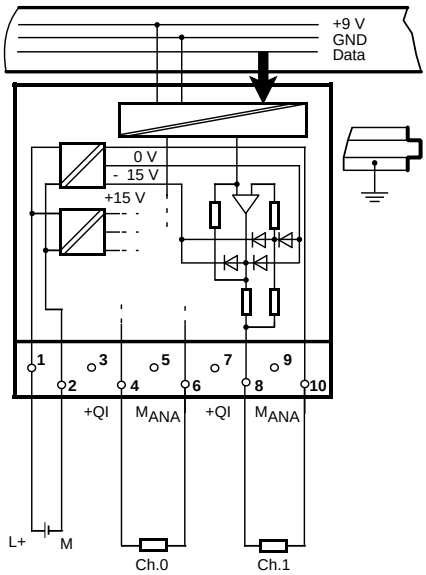
Analog Value Representation

Table A-17. Analog Output Module 470-8MA12 (Bipolar Fixed-Point Number)

Units	Output Values in V	High Byte	Low Byte	Range
1280	12.5	0 1 0 1 0 0 0 0	0 0 0 0 x x x x	Overrange
1025	10.0098	0 1 0 0 0 0 0 0	0 0 0 0 x x x x	
1024	10.0	0 1 0 0 0 0 0 0	0 0 0 0 x x x x	
512	5.0	0 0 1 0 0 0 0 0	0 0 0 0 x x x x	
1	0.0098	0 0 0 0 0 0 0 0	0 0 0 0 x x x x	
0	0	0 0 0 0 0 0 0 0	0 0 0 0 x x x x	
- 1	- 0.0098	1 1 1 1 1 1 1 1	1 1 1 1 x x x x	Nominal range
- 512	- 5.0	1 1 1 0 0 0 0 0	0 0 0 0 x x x x	
- 1024	- 10.0	1 1 0 0 0 0 0 0	0 0 0 0 x x x x	
- 1025	- 10.0098	1 0 1 1 1 1 1 1	1 1 1 1 x x x x	
- 1280	- 12.5	1 0 1 1 0 0 0 0	0 0 0 0 x x x x	Overrange

Analog Output Module 2 x±20 mA (6ES5 470-8MB12)





Technical specifications

Output range (rated values)		±20 mA
Number of outputs		2
Galvanic isolation		yes (outputs to grounding point and between outputs)
Load resistance		300 k
Connection method		two-wire connection
Digital representation of output signal		11 bits + sign (1024 units = rated value)
Measured value representation		two's complement (left-justified)
Conversion time (0 to 100%)	max.	0.15 ms
Permissible overload		25%
Open-circuit voltage		±15 V
Permissible voltage difference to ground and between outputs	max.	75 V DC/60 V AC
Basic error limits (operational error limits at 25°C, referred to output range)		±0.3%
Operational error limits (0 to 60 °C, referred to output range) (32 to 140 °F)		±0.6%
Length of cable - shielded	max.	200 m (660 ft.)
Supply voltage L+ - rated value - ripple V_{PP} - permissible range (ripple included)		24 V DC 3.6 V 20 to 30 V
Rated insulation voltage (+9 V to ⏚) - insulation group - tested with		12 V AC 1 x B 500 V AC
Rated insulation voltage (Output to L +, between outputs, output to + 9V) - insulation group - tested with		60 V AC 1 x B 500 V AC
Current consumption - from + 9 V (CPU) - from L +	typ. typ.	170 mA 130 mA
Power loss of the module	typ.	3.8 W
Weight	approx.	290 g (10 oz.)

Legend:
QI: Analog ouptput "current"

Connection of the Module

Figure A-13 shows how to connect loads to the current outputs of the module.

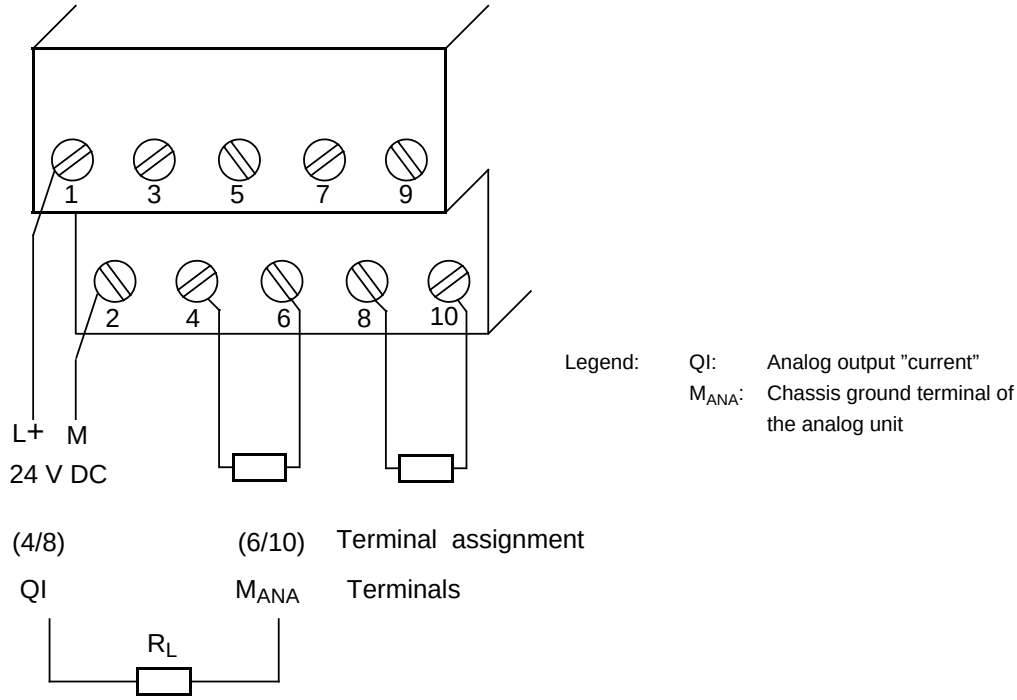


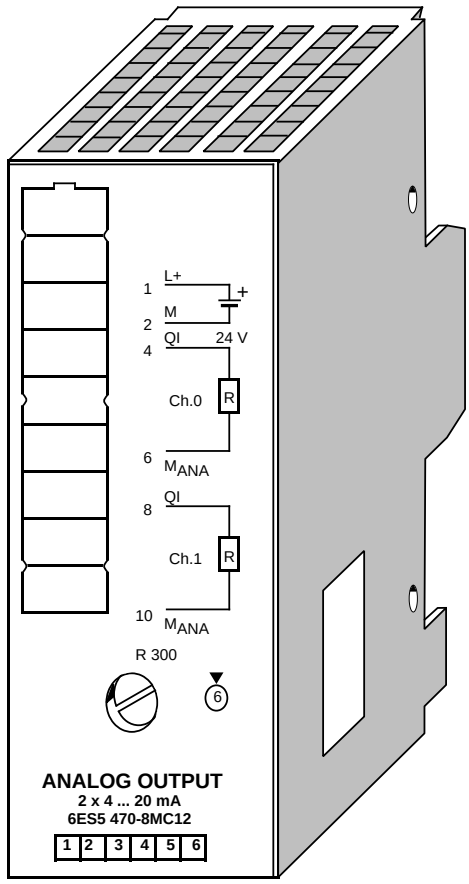
Figure A-13. Connection via a Two-Wire Circuit (6ES5 470-8MB12)

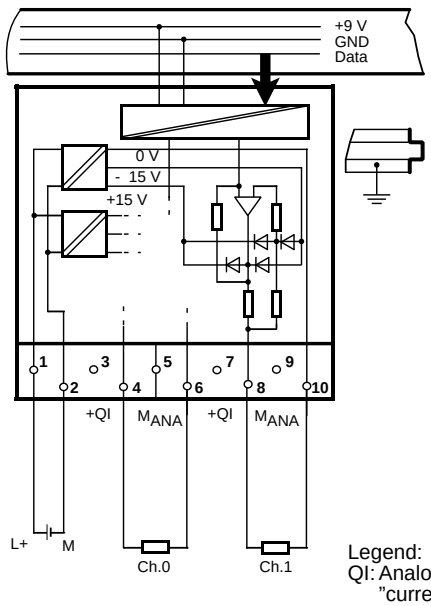
Analog Value Representation

Tabelle A-18. Analog Output Module 470-8MB12 (Bipolar Fixed-Point Number)

Units	Output Values in mA	High Byte	Low Byte	Range
1280	25.0	0 1 0 1 0 0 0 0	0 0 0 0 x x x x	Overrange
1025	20.0195	0 1 0 0 0 0 0 0	0 0 0 0 x x x x	
1024	20.0	0 1 0 0 0 0 0 0	0 0 0 0 x x x x	
512	10.0	0 0 1 0 0 0 0 0	0 0 0 0 x x x x	
1	0.0195	0 0 0 0 0 0 0 0	0 0 0 0 x x x x	
0	0	0 0 0 0 0 0 0 0	0 0 0 0 x x x x	
- 1	- 0.0195	1 1 1 1 1 1 1 1	1 1 1 1 x x x x	Nominal range
- 512	- 10.0	1 1 1 0 0 0 0 0	0 0 0 0 x x x x	
- 1024	- 20.0	1 1 0 0 0 0 0 0	0 0 0 0 x x x x	
- 1025	- 20.0195	1 0 1 1 1 1 1 1	1 1 1 1 x x x x	
- 1280	- 25.0	1 0 1 1 0 0 0 0	0 0 0 0 x x x x	Overrange

Analog Output Module 2 x 4 to 20 mA (6ES5 470-8MC12)





Legend:
QI: Analog output "current"

Technical specifications

Output range (rated value)		4 to 20 mA
Number of outputs		2
Galvanic isolation		yes (outputs to grounding point and between outputs)
Load resistance		300
Capacitive load incl. cable capacitance		< 100 nF
Connection method		two-wire connection
Digital representation of output signal		11 bits + sign (1024 units = rated value)
Measured value representation		two's complement (left-justified)
Conversion time (0 to 100%)	max.	0.15 ms
Permissible overload		25%
Open-circuit voltage		± 15 V
Permissible voltage difference to central ground point and between outputs	max.	75 V DC/60 V AC
Basic error limits (operational error limits at 25°C, referred to output range)		±0.2%
Operating error limits (0 to 60 °C, referred to output range) (32 to 140 °F)		±0.6%
Length of cable - shielded	max.	200 m (660 ft.)
Supply voltage L+ - rated value		24 V DC
- ripple V_{PP}		3.6 V
- permissible range (ripple included)		20 to 30 V
Rated insulation voltage (+9 V to ⏚) - insulation group - tested with		12 V AC 1 x B 500 V AC
Rated insulation voltage (outputs to L+, between outputs, output to + 9 V) - insulation group - tested with		60 V AC 1 x B 500 V AC
Current consumption - from + 9 V (CPU)	typ.	170 mA
- from L+	typ.	130 mA
Power loss of the module	typ.	3.8 W
Weight	approx.	290 g (10 oz.)

Connection of the Module

Figure A-14 shows how to connect loads to the current outputs of the module.

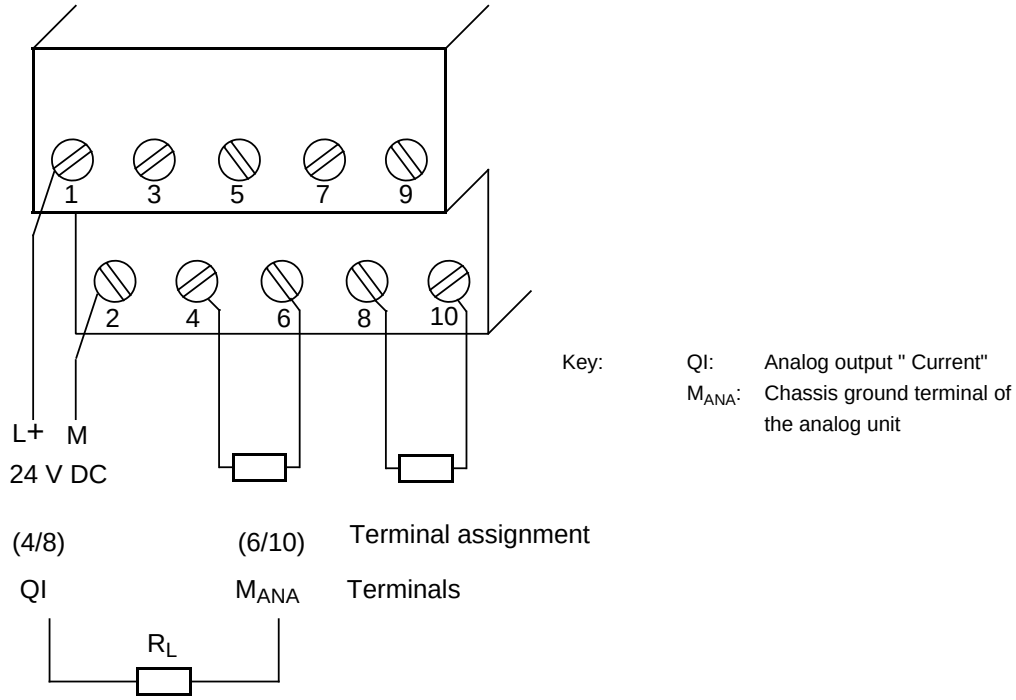


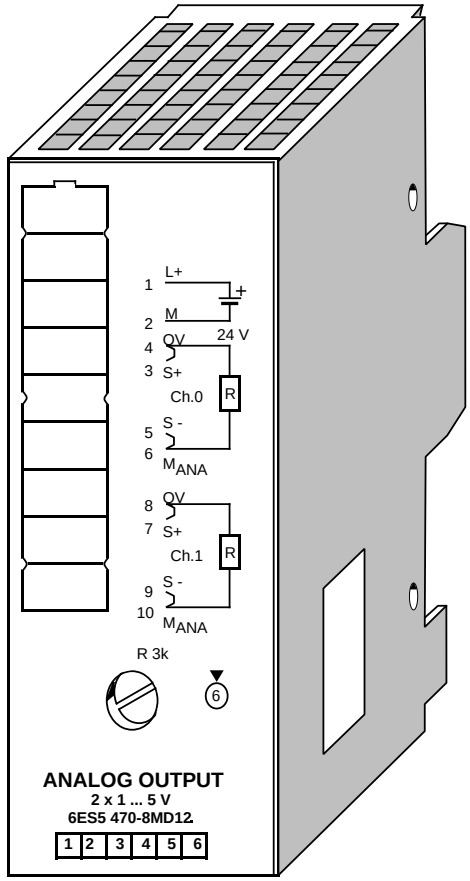
Figure A-14. Connection via a Two-Wire Circuit (6ES5 470-8MC12)

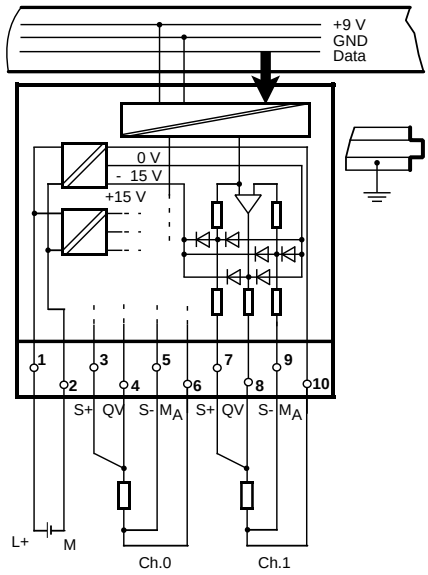
Analog Value Representation

Table A-19. Analog Output Module 470-8MC12 (Unipolar)

Units	Output Values in mA	High Byte	Low Byte	Range
1280	24.0	0 1 0 1 0 0 0 0	0 0 0 0 x x x x	Overrange
1025	20.016	0 1 0 0 0 0 0 0	0 0 0 1 x x x x	
1024	20.0	0 1 0 0 0 0 0 0	0 0 0 0 x x x x	
512	12.0	0 0 1 0 0 0 0 0	0 0 0 0 x x x x	Nominal range
1	4.016	0 0 0 0 0 0 0 0	0 0 0 1 x x x x	
0	4.0	0 0 0 0 0 0 0 0	0 0 0 0 x x x x	
- 1	3.984	1 1 1 1 1 1 1 1	1 1 1 1 x x x x	
- 256	0.0	1 1 1 1 0 0 0 0	0 0 0 0 x x x x	Overrange
- 512	- 4.0	1 1 0 0 0 0 0 0	0 0 0 0 x x x x	
- 1024	- 12.0	1 1 0 0 0 0 0 0	0 0 0 0 x x x x	
- 1280	- 16.0	1 0 1 1 0 0 0 0	0 0 0 0 x x x x	

Analog Output Module 2 x 1 to 5 V (6ES5 470-8MD12)





Technical specifications

Output range (rated value)		1 to 5 V
Number of outputs		2
Galvanic isolation		yes (outputs to grounding point and between outputs)
Load resistance	min.	3.3 k
Connection method		two- or four-wire connection
Digital representation of output signal		11 bits + sign (1024 units=rated value)
Measured value representation		two's complement (left-justified)
Conversion time (0 to 100%)	max.	0.15 ms
Permissible overload		25%
Short-circuit protection		yes
Short-circuit current		±30 mA
Permissible voltage difference to central ground point and between outputs	max.	75 V DC/60 V AC
Basic error limits (operational error limits at 25°C, referred to output range)		±0.2%
Operating error limits (0 to 60 °C, referred to output range) (32 to 140 °F)		±0.6%
Length of cable - shielded	max.	200 m (660 ft.)
Supply voltage L+ - rated value - ripple V_{PP} - permissible range (ripple included)		24 V DC 3.6 V 20 to 30 V
Insulation rating		VDE 0160
Rated insulation voltage (+9 V to ⏚) - insulation group - tested with		12 V AC 1 x B 500 V AC
Rated insulation voltage (outputs to L+, between outputs, output to +9 V) - insulation group - tested with		60 V AC 1 x B 500 V AC
Current consumption - from +9 V (CPU) - from L+	typ. typ.	170 mA 100 mA
Power loss of the module	typ.	3.1 W
Weight	approx.	290 g (10 oz.)

Connection of the Module

Figure A-15 shows how to connect loads to the voltage outputs of the module.

The sensor lines (S+ and S-) must be directly connected to the load, so that the voltage is measured and regulated directly at the load. In this manner, voltage drops of up to 3 V per line can be compensated for.
The sensor lines can be left out if the resistances of the QV and M lines are negligible compared to the load resistance. In such a case, connect terminal S + to terminal QV, and terminal S to M_{ANA}.

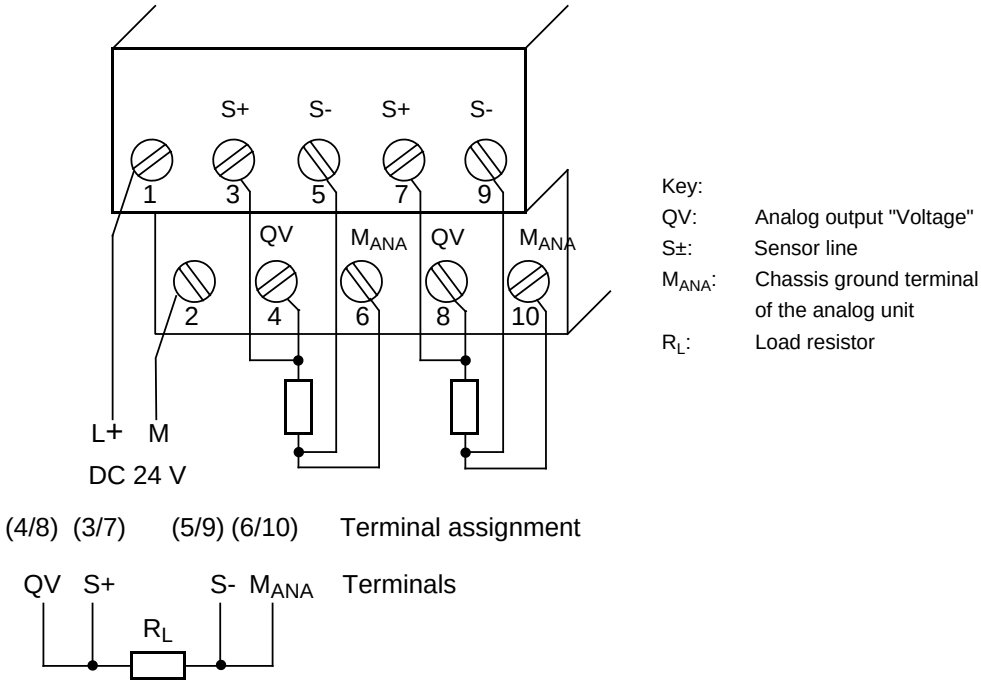


Figure A-15. Load Connection via a Four-Wire Circuit (6ES5 470-8MD12)

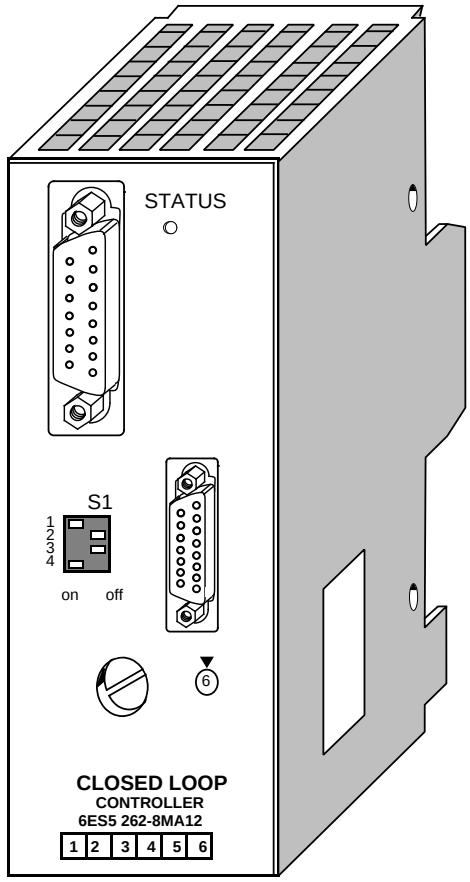
Analog Value Representation

Table A-20. Analog Output Module 470-8MD12 (Unipolar)

Units	Output Values in V	High Byte	Low Byte	Range
1280	6.0	0 1 0 1 0 0 0 0	0 0 0 0 x x x x	Overrange
1025	5.004	0 1 0 0 0 0 0 0	0 0 0 1 x x x x	
1024	5.0	0 1 0 0 0 0 0 0	0 0 0 0 x x x x	
512	3.0	0 0 1 0 0 0 0 0	0 0 0 0 x x x x	Nominal range
1	1.004	0 0 0 0 0 0 0 0	0 0 0 1 x x x x	
0	1.0	0 0 0 0 0 0 0 0	0 0 0 0 x x x x	
- 1	- 0.996	1 1 1 1 1 1 1 1	1 1 1 1 x x x x	
- 256	0.0	1 1 1 1 0 0 0 0	0 0 0 0 x x x x	Overrange
- 512	- 1.0	1 1 0 0 0 0 0 0	0 0 0 0 x x x x	
- 1024	- 3.0	1 1 0 0 0 0 0 0	0 0 0 0 x x x x	
- 1280	- 4.0	1 0 1 1 0 0 0 0	0 0 0 0 x x x x	

A.2.6 Function Modules

IP 262 Closed-Loop Control Module **(6ES5 262-8MA12)**
(6ES5 262-8MB12)



IP 262 Closed-Loop Control Module (continued) (6ES5 262-8MA12) (6ES5 262-8MB12)

Technical Specifications

Controller

Total cycle time (equals scan time)	100 to 200 ms
Resolution of the open-loop controller	5 ms at 50 Hz 4.2 ms at 60 Hz

Analog Inputs

Number of inputs	4 (suited for current, thermo-couple, or resist-ance thermometer) Voltage with external switching
Additional input for reference temperature	1 (resistance thermometer)
Galvanic isolation	no
Permissible voltage difference	
Between inputs	- 1 V to +1 V
Between inputs and central ground point	- 1 V to +1 V
Digital representation of the input signal	11 bits+sign
Current input	
Input signal range	0 to 20 mA or 4 to 20 mA
Input resistance	24.3 ±0.1%
mV Input (for thermocouple)	
Input signal range	0 to 50 mV or - 8.9 to 41.1 mV (type J, K, L, S)
Cable impedance	30 per wire
Resistance thermometer	
Start	18.49
End	219.12
Permissible cable impedance	30 per wire

Binary Inputs

Number of inputs	4
Galvanic isolation	no
Signals state ”0”	- 30 to+4.5 V or open
Signal state ”1”	+13 to +30 V (signal state invertible)
Input resistance approx.	4 k

Analog Outputs of Continuous-Action Controller (6ES5 262-8MA12)

Number of outputs	3
Galvanic isolation	no
Output signal range	0 to 20 mA or 4 to 20 mA
Maximum permissible load	600
No load voltage	(L+) - 2 V

Binary Outputs of Step-Action Controller (6ES5 262-8MB12)

Number of outputs	8
Galvanic isolation	no
Signal state ”0”	<1.5 V
Signal state ”1”	(L+) - 3.8 V
Maximum load current	100 mA short-circuit proof
Wiring method	
Programmer (PG) Operator panel (OP) SINEC-L1 network connection	front side via 15-pin subminiature D connector
Connectable are	PG 605, PG 710, PG 730, PG 750, PG 770, OP 393, OP 395
Analog and binary inputs	front side via 25-pin subminiature D connector
Analog and binary outputs	via terminal block of the bus unit
General data	
Input voltage	
Rated value	24 V DC
Permissible range	18 to 34 V DC
Permissible range with the PG 605/OP 393	18 to 27 V DC
Current consumption	
Internal (from the CPU; 9 V) approx.	20 mA
External (for 24 V; without load) approx.	180 mA
External (for 24 V; without load; with PG 605 / OP 393) approx.	340 mA
Ambient temperature	0° to 55°C (32 to 131°F)

Function

The IP 262 closed-loop control module can be used with any of the S5-90U, S5-95U, and S5-100U programmable controllers. It can be used without COM software.

The module relieves the programmable controller from closed-loop control tasks. The IP 262 also works with its own power supply in a stand-alone operation. The module can function independently without a programmable controller and can handle up to four closed-loop control circuits.

Two interfaces are located on the front panel of the module.

- An interface for the connection of a programmer (PG) or an operator panel (OP) or the SINEC L1 Network (under development)
- An interface for the connection of analog and binary inputs

In addition, the following items are available.

- A selector switch for each channel for current and voltage (thermocouples or PT 100)
- A status LED for RUN (a continuously lit green light), transducer malfunction (blinking light), and module malfunction (off)

Additionally, the module offers the following inputs.

- 4 analog inputs for direct feed of setpoint and actual values
- 4 binary inputs for control variables

The module is well-suited to take over control-loop tasks in the area of industrial processing technology, for example, temperature control, pressure and flow control, continuous injection functions, and non-time-critical closed-loop speed controls.

Modules

There are two IP 262 modules.

- ... - 8MA 12 with 3 analog outputs for continuous controllers with analog output signals
- ... - 8MB12 with 8 binary outputs for continuous controllers with pulse time interval signals for step-action controllers

Installation

- The closed-loop control module is plugged into a bus unit like any other input or output module (see Chapter 5).
- With the S5-95F, a maximum of 8 closed-loop control modules can be plugged into slots 0 through 7.
- The connections for power supply and the analog and binary output signals are located on the terminal block of the bus unit.
- The analog and binary inputs are connected to the module with a 25-pin sub-D female connector.

Addressing

The module is addressed like a four-channel analog module.

Operating Modes

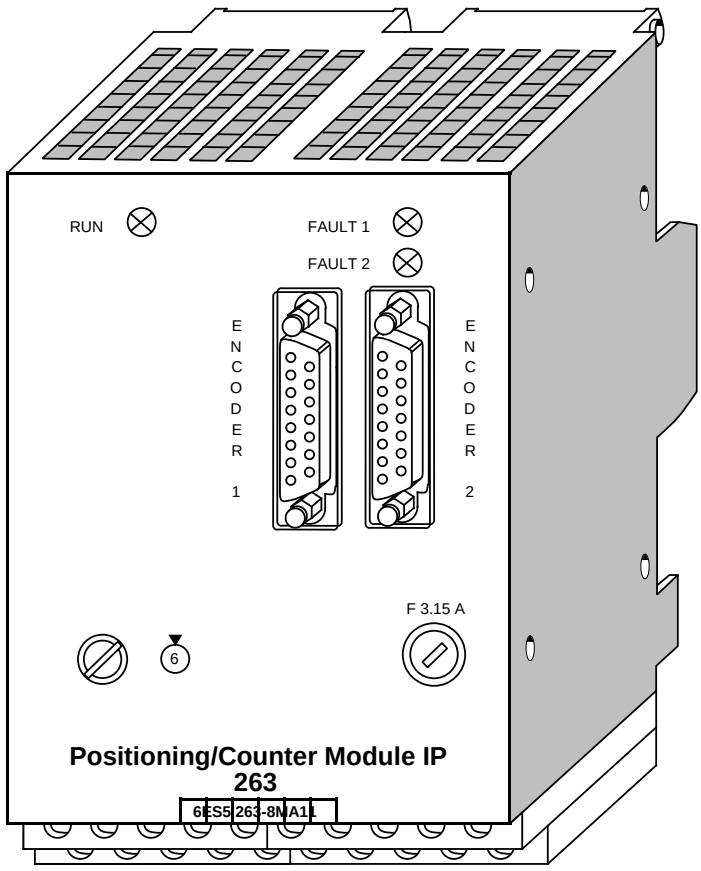
Since transducers and sensors are directly wired to the module, the module can work independently from a programmable controller in stand-alone operation, provided that the setpoints and the 24-V power supply voltage are fed directly to the IP 262. This means that the module executes the control and the output of the control value and can work alone or be controlled via the SINEC L1 by a master unit.

Besides this, the IP 262 has its own back-up, which means that the module can continue to work alone in the event the master CPU (e.g., S5-135U with R64) fails. It uses the last setpoint received from the CPU or the predefined back-up setpoint. Two operating modes are possible.

- DDC-Operation (Direct Digital Control):
  The control is executed entirely from the CPU and the IP only outputs the manipulated variable. If the CPU fails, the module can continue to control independently with a predefined back-up setpoint.

- SPC-Operation (Setpoint Control):
  The module receives only the setpoint from the CPU; the control is carried out independently of the CPU. If the CPU fails, the IP continues to control using the last setpoint received from the CPU. It is also possible to use a predefined back-up setpoint here.

IP 263 Positioning Module **(6ES5 263-8MA11)**



IP 263 Positioning Module (continued) (6ES5 263-8MA11)

Technical Specifications

Encoders

Position decoder	incremental, absolute (SSI interface)
Maximum traversing range	
- with incremental encoders	2^{24} increments
- with absolute encoders	8192 increments per revolution ×2048 revolutions
Signal voltages	
- Differential inputs	5 V to RS 422
- Asymmetrical inputs	24 V (only incremental encoders)
Supply votlage for encoders (short-circuit-proof, no overload)	5 V/300 mA 24 V/300 mA
Input frequency and cable length	
Symmetrical encoders (5 V signals):	
- with 5 V encoder supply	max. 200 kHz for 32 m (105 ft.) cable, shielded
- with 24 V encoder supply	max. 200 kHz for 100 m (328 ft.) cable, shielded
Asymmetrical encoders (24 V signale):	max. 100 kHz for 25 m (82 ft.) cable, shielded max. 25 kHz for 100 m (328 ft.) cable, shielded
Data transmission rate and cable length with absolute encoders	62.5 kHz (selectable in steps) 125 kHz (160 m/ 525 ft. shielded) 250 kHz 500 kHz 1 MHz (32 m/ 105 ft. shielded)
Input signals	
- Incremental	2 pulse trains displaced by 90° 1 zero pulse
- 24 V initiator (BERO)	1 pulse train
- SSI	Absolute value
Input currents	to RS 422
- 5 V	typ. 5 mA
- 24 V	

Digital Inputs

Input voltage range	- 3 V to + 30 V
Galvanic isolation	no
0 signal	- 3 V to +5 V
1 signal	+13 V to+30 V
Permissible zero-signal current at 0 signal	1.1 mA
Input current at 24 V	typ. 5 mA

Other: If the digital inputs are used, they must always be connected to a defined potential (0 V, 24 V) and must not be kept open.

Digital Outputs

Output voltage range	+20 V to+30 V
Galvanic isolation	no
Output current at 1 signal	max. 500 mA
Short-circuit protection	Short-circuit-proof output
Cable length, shielded	max. 100 m (328 ft.)

Supply Voltage

Logic voltage from 24 V supply produced with switched-mode power supply	4.9 V to 5.1 V
Current consumption from 24 V without outputs and encoder	typ. 120 mA
Undervoltage monitoring	$V_{internal} < 4.65$ V
Power Loss	typ. 4 W

A separate manual is available for the IP 263 positioning module. It can be ordered under the order number 6ES5 998-5SK11.

The IP 263 is suitable for positioning of two independent axes.

Assignments of Outputs

The IP 263 is a two-channel module: 4 digital outputs are assigned to each channel for the control of drives;

- Rapid traverse
- Creep speed
- Anti-clockwise rotation
- Clockwise rotation

Both incremental and absolute encoders (SSI - synchronous serial interface) can be connected for actual position encoding.

They transmit the machine data, such as

- Software limit switches
- Resolution
- Cutoff difference
- Switchover difference
- Zero-speed control

The syntax for the data block which has to be created for this purpose is simple and is described in the manual.

Positioning

The only thing that remains to be done is to specify the desired target and then the module is ready for the positioning procedure.

The IP 263 then carries out positioning automatically. When the target has been reached, it sends a message to the IM 318-B interface module and thus to the CPU.

Figure A-16 shows the positioning procedure with the IP 263: After the start, a rapid traverse towards the target takes place first. When the switchover/cutoff point has been reached, a switch-over to creep speed or cutoff takes place. Afterwards, the IP 263 monitors approach of the target. When the axis has reached the target range, a signal is sent to the IM 318-B interface module.

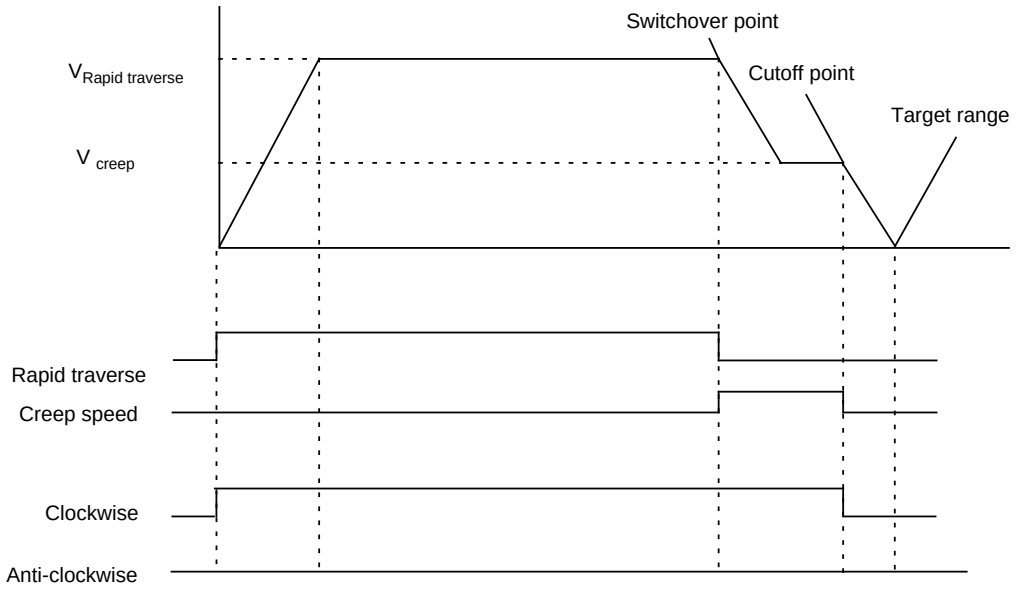


Figure A-16. Positioning with the IP 263

During reference point travel, the digital input of the module senses the speed reducing cam (reference point switch).

In the ”Length measurement” operating mode, the module senses encoder pulses as long as this input has a ”1” signal.

Installation

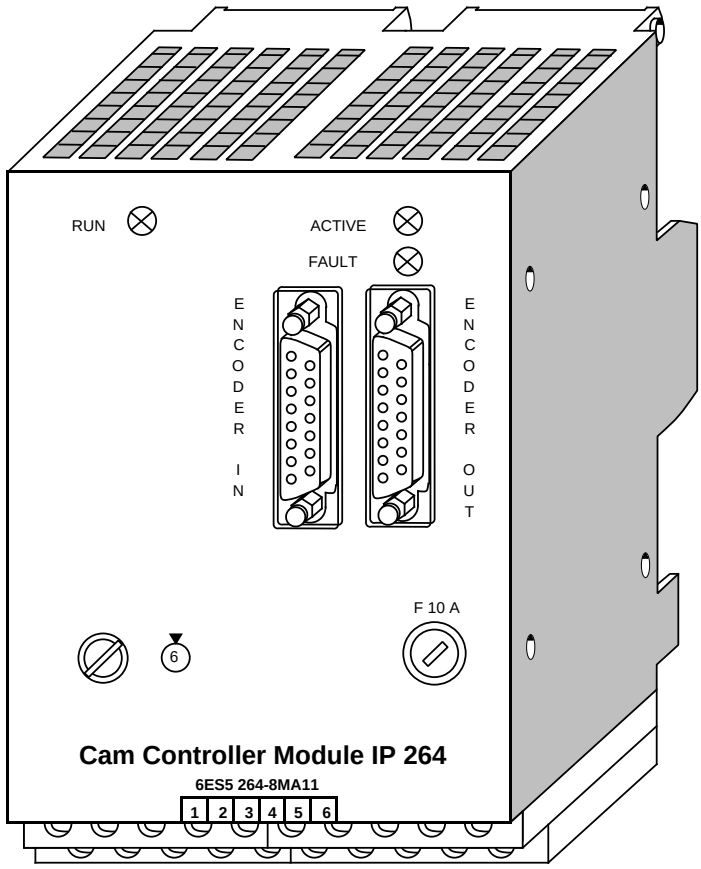
As other I/O modules, the IP 263 is mounted on the bus unit (see Chapter 5).

- With the S5-95U, a maximum of 4 positioning modules can be plugged in only at slots 0 through 7.

Addressing

The IP 263 is addressed like a 4-channel analog module.

IP 264 Electronic Cam Controller Module **(6ES5 264-8MA11)**



IP264 Electronic Cam Controller Module (6ES5 264-8MA11)

Technical Specifications

Encoders

Actual value sensing	incremental, absolute (SSI interface)
Maximum traversing range	
- with incremental encoders	2^{16} increments
- with absolute encoders	2^{16} encoders
Signal voltages	
- Differential inputs	5 V to RS 422
- Asymmetrical inputs	24 V (only incremental encoders)
Supply voltage for encoders (short-circuit-proof, no overload)	5 V/300 mA 24 V/300 mA
Input frequency and cable length	
Symmetrical encoders (5 V signals):	
- with 5 V encoder supply	max. 200 kHz for 32 m (105 ft.) cable, shielded
- with 24 V encoder supply	max. 200 kHz for 100 m (328 ft.) cable, shielded
Asymmetrical encoders (24 V signal):	max. 100 kHz for 25 m (82 ft.) cable, shielded max. 25 kHz for 100 m (328 ft.) cable, shielded
Data transmission rate and cable length with absolute encoders (selectable in steps)	125 kHz (160 m/ 525 ft. shielded) 250 kHz 500 kHz 1 MHz (32 m/ 105 ft. shileded)
Input signals	
- Incremental	2 pulse trains displaced by 90° 1 zero pulse
- 24 V initiator (BERO)	1 pulse train
- SSI	Absolute value
Input currents	to RS 422
- 5 V	typ. 5 mA
- 24 V	

Digital Inputs

Input voltage range	-3 V to + 30 V
Galvanic isolation	no
0 signal	- 3 V to +5 V
1 signal	+13 V to+30 V
Permissible zero-signal current at 0 signal	1.1 mA
Input current at 24 V	typ. 5 mA

Other: If the digital inputs are used, they must always be connected to a defined potential (0 V, 24 V) and must not be kept open.

Digital Outputs

Output voltage range	+20 V to+30 V
Galvanic isolation	no
Output current at 1 signal	max. 300 mA
Short-circuit protection	Short-circuit-proof output
Cable length, shielded	max. 100 m (328 ft.)

Supply Voltage

Logic voltage from 24 V supply produced with switched-mode power supply	4.9 V to 5.1 V
Current consumption from 24 V without outputs and sensors	typ. 120 mA
Undervoltage monitoring	$V_{internal} < 4.65$ V

Power Loss typ. 4 W

Module Cycle Time (incl. dead-time compensation)

Separate cam programs with max. 32 cams each for forwards and backwards (incl. dead-time compensation)	57.6 µs
"Common" cam program with max. 32/64 cams for forwards and backwards	57.6/115.2 µs

A separate manual is available for the electronic cam controller. It can be ordered under the order number 6ES5 998-5SL11.

The IP 264 can be used both for rotary and linear axes.

The IP 264 electronic cam controller makes electronic processing of cams economical even for applications in the lower performance range.

32 cams which can be allocated as desired to 16 tracks have a switching accuracy of better than 1 degree at 2400 revolutions per minute. This corresponds to a response time of less than 60 µs. For applications with low precision requirements it is even possible to program 64 cams.

It is also possible to integrate 32 cams each into a cam program for ”forwards” and a cam program for ”backwards”. Switchover between these two programs is carried out by automatic direction sensing of the IP 264 or it is controlled by the SIMATIC S5.

All cams can be defined either as path-path cams or as path-time cams.

Dead Time Compensation

Through the speed-dependent, dynamic shift, each individual cam compensates the dead time of the actuator connected (e.g. pneumatic valve) at a scanning rate of 60 µs. This enables the utmost accuracies to be achieved even at changing drive speeds.

Direct Process Connection

In order to be able to pass on the short response time of the IP 264 directly to the process, a digital output (24 V, 0.3 A) is available on the module for each track. Generally, the units to be controlled can be connected directly. Auxiliary contactors are required only for actuators with a higher current consumption.

The sensors to be connected can be incremental encoders, absolute SSI encoders (SSI= synchronous serial interface) or simple 24 V signal sensors (e.g. BEROs). The sensor data can be looped through to further modules via the additional sensor output, without separating the sensor cables mechanically or using additional fan-out units.

Installation

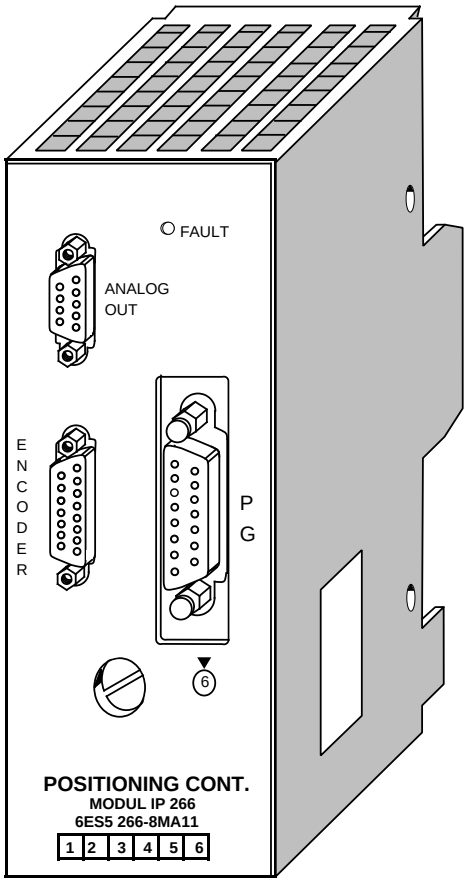
As other I/O modules, the IP 263 is mounted on the bus unit (see Chapter 5).

- With the S5-95U, a maximum of 4 positioning modules can be plugged in only at slots 0 through 7.

Addressing

The IP 264 is addressed like a 4-channel analog module.

IP 266 Positioning Module (6ES5 266-8MA11)



Technical Specifications

Analog Output		
Output signal range		±10 V
Digital signal representation		13 bits plus sign
Short-circuit proof		Yes
Reference potential of the analog output signal		analog ground of the power section
Cable length shielded	max.	32 m (105 ft)
Pulse Input		
Position decoder		incremental
Traverse range		±32767.999 mm/ 0.1 inch/grd
Input voltage for the tracks - differential inputs - asymmetrical inputs		 5 V/RS 422 24 V/typ. 7.3 mA
Supply voltage for the sensor (short-circuit proof)		5 V/350 mA 24 V/350 mA
Input Frequency and Cable Length		
Symmetrical sensors (5 V)	max. max.	500 kHz, 30 m (98 ft) shielded cable length
Asymmetrical sensors (24 V)	max.	100 kHz at 25 m cable length shielded
	max.	25 kHz at 100 m cable length shielded
Inupt Signals		2 pulse series 90 degrees out of phase, 1 zero pulse
Digital Inputs		
Input voltage range		±30 V
Galvanic isolation		no
”0” signal		- 30 V to+5 V
”0” signal		13 V to 30 V
Permissible zero signal		1.5 mA
current at ”0” signal 24 V		7.3 mA
Digital Outputs		
Output voltage range		20 V to 30 V
Galvanic isolation		no
Max. output current at ”1” signal		100 mA
Short-circuit protector		short-circuit proof output
Cable length shielded	max.	100 m (330 ft.)
Supply voltage		
Logic voltage from 24 V ext. supply produced with switched-mode power supply		4.7 V to 5.5 V
Current consumption from 24 V supply without outputs and 24 V sensor	typ.	180 mA

Because of its performance capability and the complexity of its description, the IP 266 has its own manual. The order number is: 6ES5 998-5SC21.

As an ”intelligent I/O module”, it allows you to use open-loop as well as closed-loop control positioning.

The positioning operations are processed independently of the execution times of the user programs in the programmable controller. Thus the CPU is not burdened with positioning jobs constantly being processed.

Operation Principle of the IP 266

The IP 266 enables you to control the positioning operation of your drive exactly. The module delivers a voltage setpoint in the range of ±10 V via an analog output for the control of a power section for servo motors.

The IP 266 needs exact data about your drive system in order to calculate speed, acceleration, or traverse residual distances. This data can be stored in an EEPROM that is permanently installed in the programmable controller. By using its own start-up routine, this data can be accessed immediately after you switch on the programmable controller and can be processed directly.

The IP 266 allows you to select between a linear axis and a circular axis. You can also select the unit of measurement for processing the data: either [mm], [in.] or [deg].

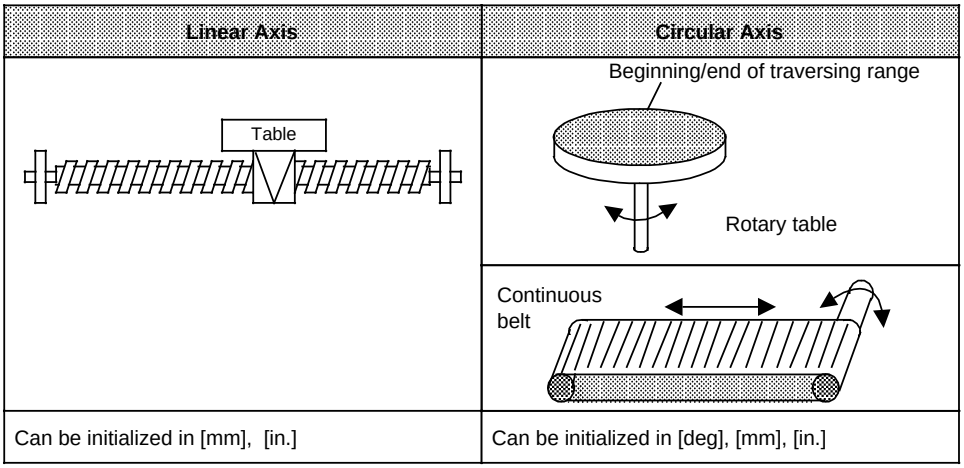


Figure A-17. Processed Units of Measurement for Circular Axis and Linear Axis

Besides purely traversing movements, other operating modes allow offset generation of axis coordinates or drift compensation in the system.
In addition, the IP 266 offers operating modes to read data such as positioning actual value or residual traversing distances.

In order to use the IP 266 in an automatic manufacturing process, it is possible to combine individual traversing applications, positioning corrections, offsets or dwell times in a ”traversing program”. These traversing programs can be called up via two special operating modes and processed automatically or semi-automatically.

Such a traversing program can be created by using the ”learning capable” ”Teach-in-mode” for positioning applications. The information from single positioning applications can be stored at the end of an operation in a traversing program.

Positioning

For the positioning operation, the IP 266 calculates the setpoint from the selected end data and velocity data in conjunction with the programmed machine data. The actual value follows the selection. The deviation (following error) which occurs reaches a constant value after the short start-up phase and must reach zero at the end of the positioning operation.

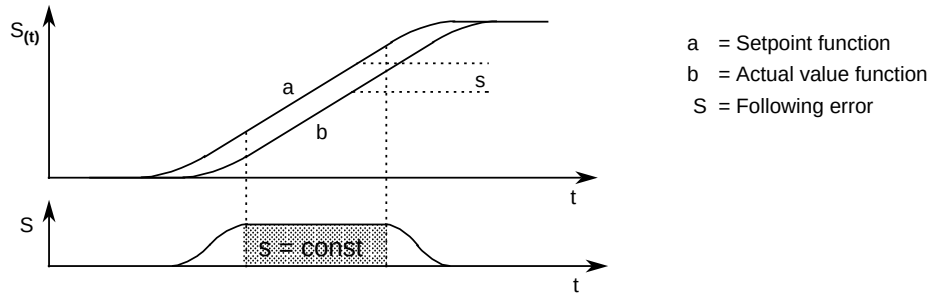


Figure A-18. Course of a Following Error during a Positioning Operation

Overview of the Operation Modes

Table A-21. Designation of the Operating Modes

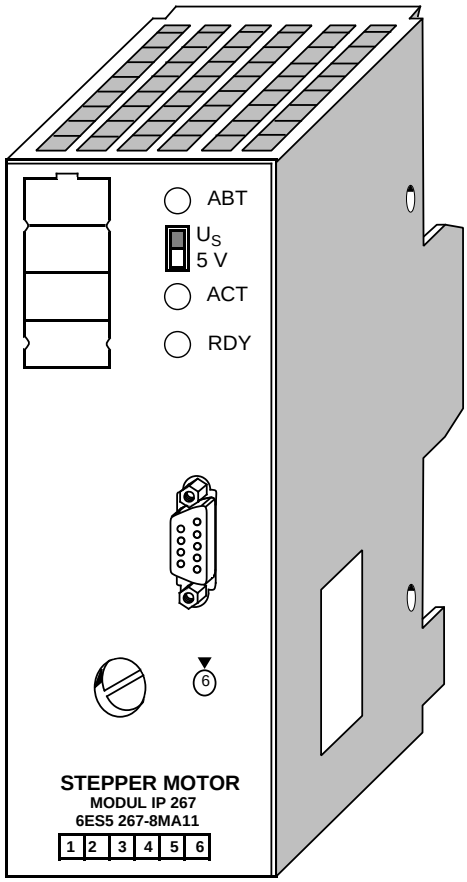
Description		
JOG 1 JOG 2 CONTROLLED JOG FOLLOW-UP MODE REFERENCE POINT INCREMENTAL ABSOLUTE INCREMENTAL RELATIVE AUTOMATIC AUTOMATIC SINGLE BLOCK TEACH-IN ON	TEACH-IN OFF ZERO OFFSET ABSOLUTE ZERO OFFSET RELATIVE CLEAR ZERO OFFSET TOOL OFFSET ON TOOL OFFSET OFF ACKNOWLEDGE ERROR DRIFT COMPENSATION ON DRIFT COMPENSATION OFF WRITE MACHINE DATA	RAM EEPROM FREE SPEED DISPLAY FREE SPEED SETTING MONITORING OFF MONITORING ON READ ACTUAL POSITION READ FOLLOWING ERROR READ DISTANCE TO GO SYNCHRONIZE IP

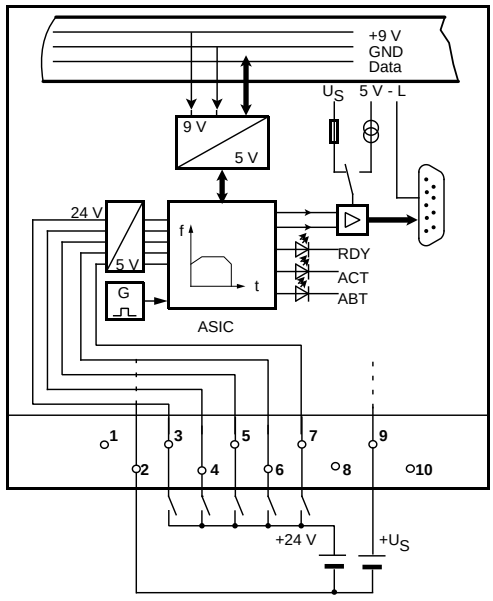
The COM 266 software package offers user friendly operation and parameter settings. The IP 266 exchanges all data with the programmable controller via a serial interface. All tasks written in 8-byte messages are sent to the IP 266 during the program cycle via the process output image (PIQ). The IP 266 transmits feedback messages cyclically via the process image input (PII). These messages can be about the actual value position, remaining traversing distance, or following error as well as a status byte, error byte, the current operation mode, and special data from the traversing program.

Installation

- Plug the IP 266 into a bus unit like any other I/O module.
- With the S5-95F, a maximum of 8 positioning modules can be plugged in only at slots 0 to 7.
- Connect the external switches to the digital inputs of the IP 266 via the terminal block. These switches are used to limit the traversing range. They also allow you to intervene at any time into the processing of the module.
- The IP 266 can bypass the CPU of the programmable controller, via three digital outputs, and send signals directly to external I/Os. The controller must, however, be enabled (function signal enable controller, FUM) and must be connected to the power section of the drive.
- Connect the servo motor's power section to the 9-pin subminiature D female connector.
- Connect the incremental encoder to the left 15-pin subminiature D female connector ”ENCODER”.
- You can connect a programmer with screen to the 15-pin subminiature connector on the right side to operate the IP 266 via the COM 266 software.

IP 267 Stepper Motor Control Module (6ES5 267-8MA11)





Technical Specifications

Supply voltage (PLC BUS)		9 V
Current consumption	approx.	150 mA
Special voltage V_S		5 V to 30 V

Digital Inputs

Rated input voltage		24 V
Galvanic isolation		no
Input voltage:		
"0" signal		- 33 V to 5 V
"1" signal		13 V to 33 V
Input current	typ.	8.5 mA
Supply voltage for two-wire BEROs		22 V to 30 V

9-pin Subminiature D Interface

Output voltage with 5-V supply		
"0" signal	max.	0.4 V
"1" signal	min.	4.5 V
With special supply voltage V_S (5 V to 30 V)		
"0" signal	max.	0.4 V
"1" signal	min.	V_S - 0.4 V
Output current		20 mA (short-circuit proof)
Output frequency	max.	204 kHz
Increment number of steps	max.	2^{20} - 1 pulses/ job
Permissible cable length	max.	50 m (165 ft.) at 50 kHz (twisted pair cable)

Because of its performance capability and the complexity of its description, the IP 267 has its own manual. The order number is: 6ES5 998-5SD21. The IP 267 Stepper Motor Control Module expands the field of application as an intelligent I/O module (IP) of the S5-95F programmable controller for "closed-loop control positioning". The IP 267 controls positioning processes independently of the run time of user programs in the programmable controller. The CPU is not loaded with processing positioning jobs.

Principle of Operation

The IP 267 generates pulses for the stepper motor power section. The number of output pulses determines the length of the traversing path and the pulse frequency is a measure of the velocity. Each pulse causes the stepper motor shaft to turn through a certain angle. In the case of high-speed pulse trains, this step movement becomes a constant rotational movement. Stepper motors can reproduce all movement sequences only as long as no steps are lost. Step losses can be caused when load variations occur or when the programmed pulse trains exceed motor-specific values.

To enable the IP 267 to generate these pulse trains, the user must enter the following data.

- Configuration data: This data describes the individual stepper motors and the technical characteristics of the drive system.
- Positioning data: This data describes the individual traverse jobs and indicates the velocities, directions, and lengths of the configured paths.

The IP exchanges data with the programmable controller via the serial interface. During the program scans, all necessary information is sent from the process image output table (PIQ) to the IP 267 in 4-byte messages. The IP 267 cyclically transmits feedback signals on the remaining distance to go and various status bits to the process image input table (PII).

Using the configuration and positioning data settings, the IP 267 generates a symmetrical traverse profile consisting of an acceleration ramp, a constant velocity range and a deceleration ramp.

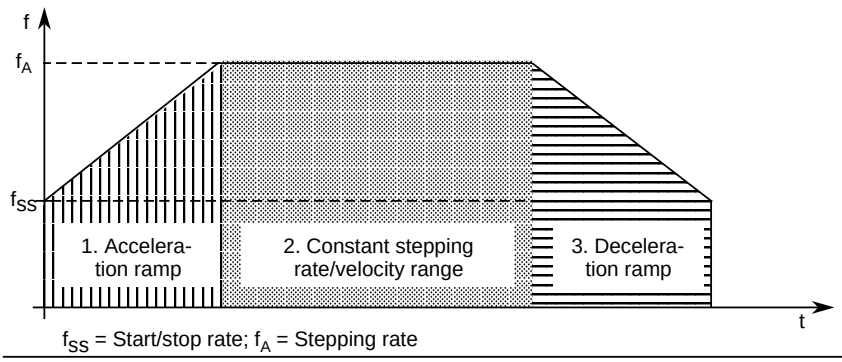


Figure A-19. Velocity Profile of the IP 267

Using a limit switch on the digital inputs, IP 267 can monitor the limits of a traversing range and stop the traversing movement when the permissible range limit is exceeded.

The activated input "external stop" causes a calculated decelerating of the traversing movement. An emergency limit switch can be installed at input "IS" (pulse inhibit). When this switch responds, the pulse output is interrupted immediately.

For a reference point approach operation, an additional switch can be connected at input REF that lies within the traversing zone. The reference point approach operation is also possible without this switch.

Status LEDs provide you with the following information.

The IP 267 is configured	RDY
Pulse outputs during a positioning operation	ACT
Interruption of the positioning operation	ABT

There are four operating modes:
STOP
START FORWARDS
START BACKWARDS
NEUTRAL

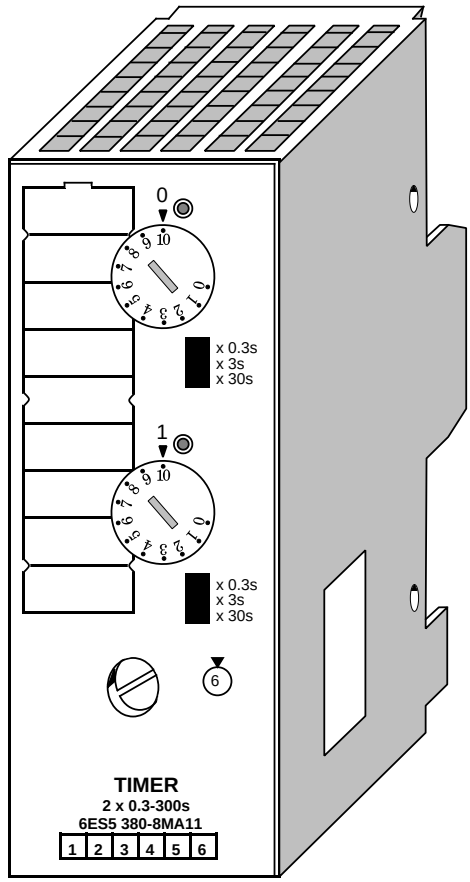
Installation

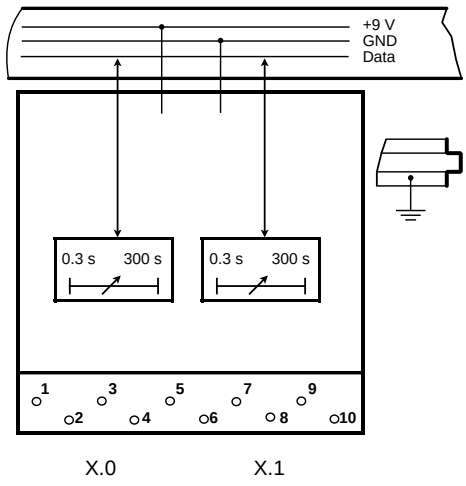
- Plug the IP 267 into a bus unit like any other I/O module.
- With the S5-95F, a maximum of 6 IP 267 modules (limited by current consumption) can be plugged in at slots 0 to 7.
- Connect the external switches to the DIs of the IP 267 via the terminal block.
- Connect the stepper motor's power section to the 9-pin subminiature D female connector.

Addressing

The IP 267 is addressed like an analog module.

Timer Module 2 x 0.3 to 300 s (6ES5 380-8MA11)

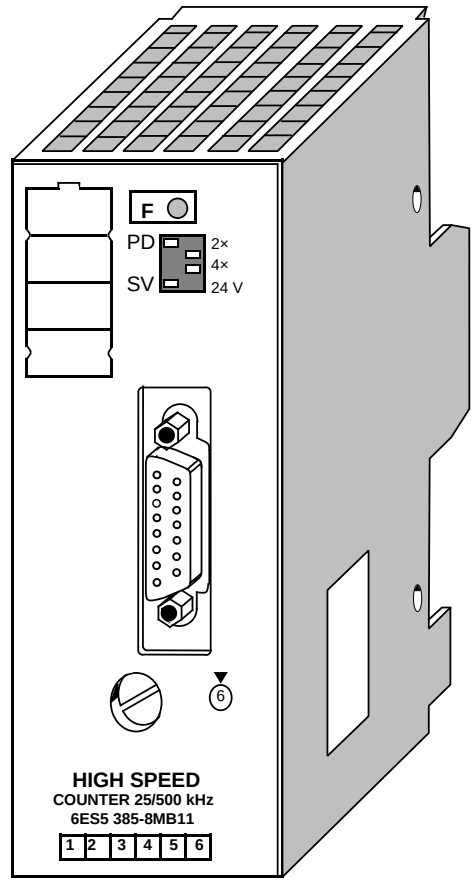


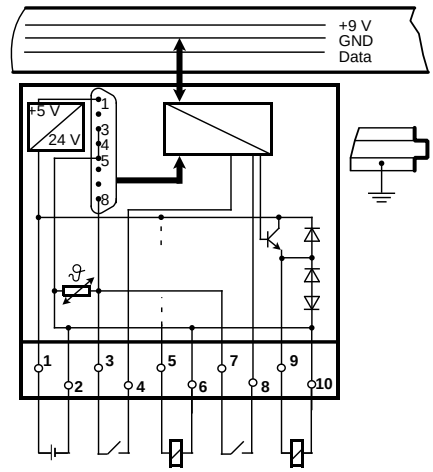


Technical Specifications

Number of timers		2
Time setting Range extension factor		0.3 to 3 s ×10, ×100
Function display		green LED
Setting error		±10 %
Reproducibility		±3%
Temperature influence		+1 %/10° C (50° F) from set time
Rated insulation voltage (+ 9V to ⏚) - insulation group - tested with		12 V AC 1×B 500 V AC
Current consumption - from +9 V (CPU)	typ.	10 mA
Weight	approx.	200 g (7 oz.)

Counter Module 25/500 kHz **(6ES5 385-8MB11)**





Counter Module 25/500 kHz (continued) (6ES5 385-8MB11)

Technical Specifications

Operating mode (switch-selectable)		
- position decoder		PD
- counter		C
Decoder inputs		1 decoder 5 V (differential input) or 1 decoder 24 V DC
Digital inputs		2; reference and enabling
Digital outputs		2; setpoints reached 1 and 2
Galvanic isolation		no
Counting range		
Operating mode		
- position decoder		two's complement (KF) - 32768 to +32767
- counter		unipolar representation (KH) 0 to 65535
Counting mode		
- position decoder		forwards/backwards
- counter		up
Setpoint input		via program
5-V Decoder Input		15-pin Cannon sub-miniature D connector
Input signals		differential signals to RS 422
- position decoder		A A-N, B B-N, R R-N
- counter		A A-N
Counting frequency	max.	500 kHz
Cable length (shielded)	max.	50 m (165 ft.)
Power supply for decoder		5 V from L+via voltage transformer
Output current	max.	300 mA, short-circuit-proof
24-V Decoder Input		15-pin Cannon sub-miniature D connector
Rated input voltage		24 V DC
Input signals		
- position decoder		A, B, R
- counter		A
Input voltage		
- "0" signal		- 33 to 5 V DC
- "1" signal		+13 to 33 V DC
Rated input current for "1" signal	typ.	8.5 mA
Counting frequency	max.	25 kHz
Cable length (shielded)	max.	100 m (330 ft.)
Power supply for decoder		24 V from L+ (PTC thermistor)
Output current	max.	300 mA, short-circuit-proof
Digital Inputs		reference and enabling
Rated input voltage		24 V DC
Input voltage		
- "0" signal		- 33 to +5 V DC
- "1" signal		+13 to 33 V DC
Rated input current for "1" signal and at 24 V	typ.	8.5 mA
Input frequency	max.	100 Hz
Inherent delay	typ.	3 ms (1.4 to 5 ms)
Cable length (unshielded)	max.	100 m (330 ft.)
Short-circuit protection (cable impedance up to 15)		
Digital Outputs		setpoints 1 and 2
Output current (resistive, inductive load)		5 mA to 0.5 A
Residual current for "0" signal	max.	0.5 mA
Switching current for lamps		0.22 A (5 W)
Limitation of inductive interrupting voltage	to	-15 V
Output voltage		
- "1" signal	min.	L+ - 2.2 V
- "0" signal	max.	3 V
Cable length (unshielded)	max.	100 m (330 ft.)
Short-circuit protection (cable impedance up to 15)		electronic
Short-circuit indication (short-circuit to M)		red LED
Supply voltage L+		
- rated value		24 V DC
- ripple V_{pp}	max.	3.6 V
- permissible range (including ripple)		20 to 30 V DC
Fuse (internal)		T 5 A
Current consumption		
- from L+without decoder supply without load		30 mA
- internal (+9 V)		70 mA
Power consumption of the module	typ.	1.9 W+total output current (I_A)×1.1 V
Weight	approx.	250 g (9 oz.)

Function

The counter module can be used as an up-counter or as an up/down counter for a position decoder.

The counting pulses are supplied by a sensor that you can connect to the 15-pin subminiature D female connector of the module. You can choose from two types of sensors that fulfill the following requirements.

- 5-V error voltages according to RS 422 (up to 500 kHz) or
- 24-V signals (up to 25 kHz)

As additional inputs, the module has an enable input and a reference input.

By using the STEP 5 program, you can assign two setpoints via the I/O bus. Once the counter status reaches one of these values, the respective output completes the circuit at the terminal block (Q0 or Q1). The status of the outputs is displayed in the diagnostic byte.

You can also read the following values by using the STEP 5 program.
- The updated count
- The diagnostic byte

You can preselect the following items on the operating mode switch.
- Function mode
- Position resolution
- Input voltage range of the sensor

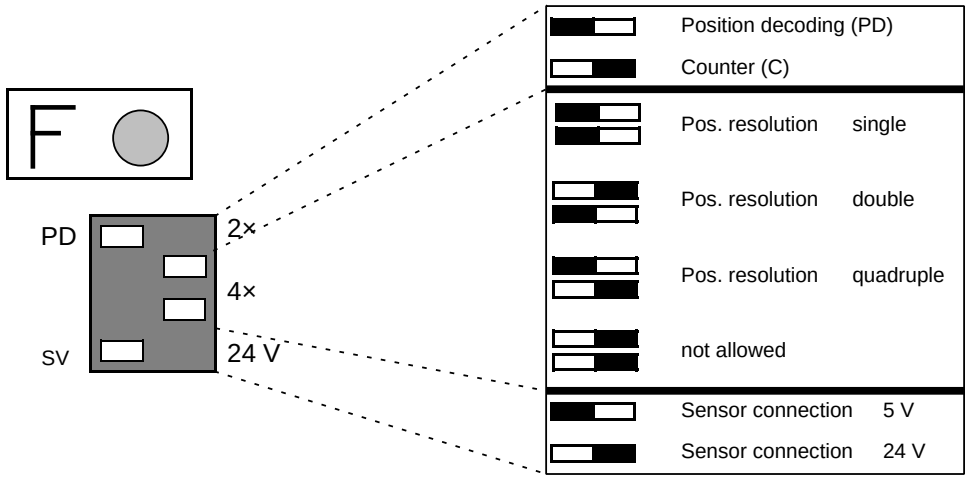
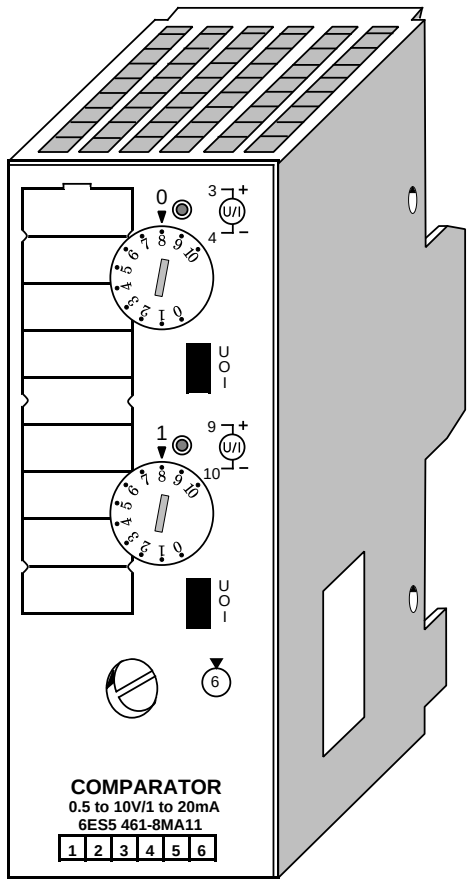
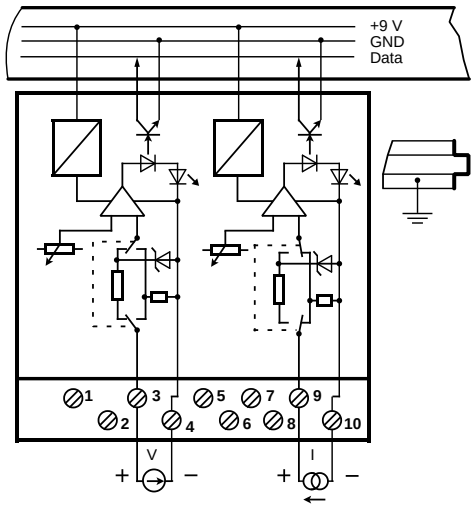


Figure A-20. Switch Positions on the Operating Mode Switch

Comparator Module 2 x 0.5 to 20 mA/0.5 to 10 V (6ES5 461-8MA11)





Technical Specifications

Channels		2
Galvanic isolation		yes
Current or voltage measurement		switch-selectable
Switch position "0"		no measuring
Display		green LED for actual value setpoint
Setpoint adjustment		with potentiometer
Setting error		±10 %
Reproducibility		±2 %
Hysteresis		10 %
"V" measuring range		0.5 to 10 V DC
Input resistance		47 k
Inherent delay	typ.	5 ms
Input voltage	max.	100 V DC (0.5 s)
"I" measuring range		0.5 to 20 mA
Input resistance		500
Overload capability		100 %
Rated insulation voltage (+9 V to measuring circuit and between measuring circuits)		30 V AC
- insulation		2×B
- tested with		500 V AC
Rated insulation voltage (+9 V to ⏚)		12 V AC
- insulation group		1×B
- tested with		500 V AC
Length of cable		
- shielded		200 m (660 ft.)
- unshielded		100 m (330 ft.)
Current consumption - from +9 V (CPU)	typ.	35 mA
Power losses of the module	typ.	0.3 W
Weight	approx.	200 g (7oz.)

Function

The module has two isolated comparators for voltage or current measurement (selector switch with positions U/0/I). When the preset value is reached, the LED of the respective channel lights up and sends a ”1” signal to the programmable controller.

The module must be removed or the measuring circuit disconnected before you select the function.

In switch position ”0”, the comparator is switched off; if scanned, a ”0” signal results.

The response threshold of the comparator is set by a selector on the front panel. The selector has scale divisions to simplify adjustment.

Installation

The comparator module is mounted on a bus unit like any other input or output module (see Chapter 5).

Wiring

See schematic diagram. Unused inputs can be left open.

Addressing

The comparator module is addressed like a 2-channel digital input module (channel ”0” or ”1”).

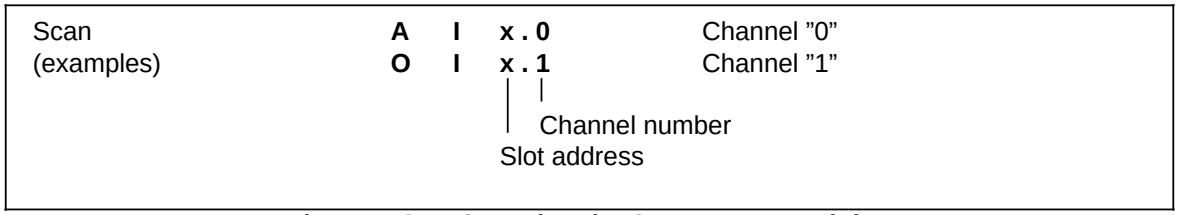
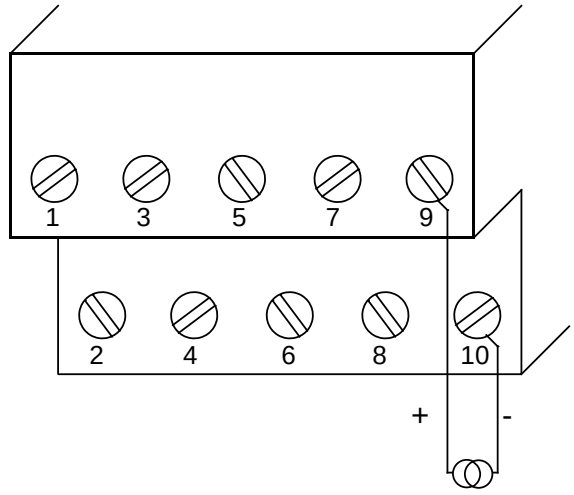


Figure A-21. Scanning the Comparator Module

Typical Application

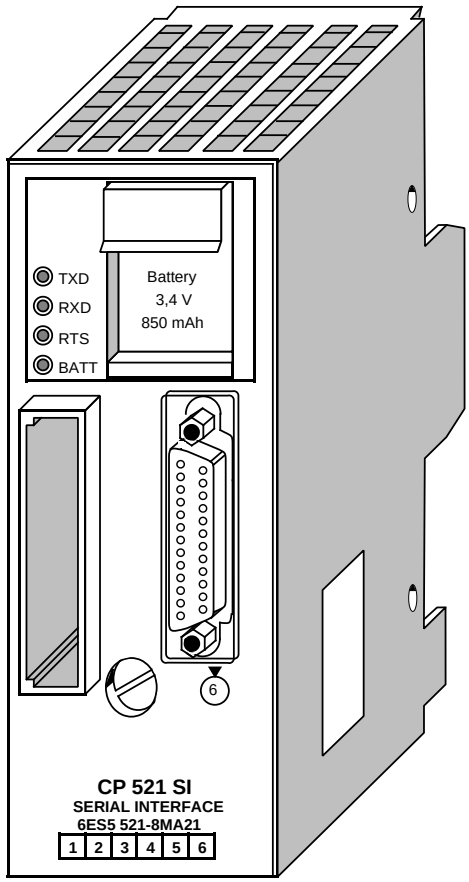
A comparator module is mounted at slot 4. The current source is connected to channel 1. If the Schmitt trigger 1 detects that the current has exceeded the preset value, output 5.1 is to be set.

Terminal Connections



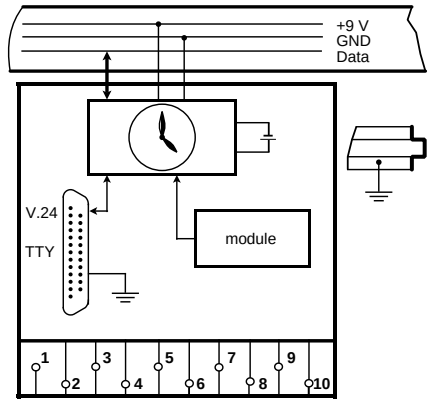
STL	Explanation
A I 4.1 = Q 5.1	As soon as the limit is reached or exceeded, input 4.1 becomes "1"; this sets output 5.1 to "1".

CP 521 SI Communications Processor (6ES5 521-8MA21)



Technical Specifications

Galvanic isolation	TTY signals are isolated
Memory submodule	EPROM/EEPROM
Serial interface	V.24/TTY passive (active)
Transmission	Asynchronous 10-bit character frame/11-bit character frame
Transmission rate	110 to 9600 baud
Permissible cable length - V.24 - TTY	 15 m (49.2 ft.) Results from: (Voltage drop on cable)+ (Receiver-typ. voltage drop 1.5 V) or (transmitter-typ. voltage drop 0.9 V) max. 1000 m (3281 ft.)
LED displays - TxD (green) - RxD (green) - RTS (green) - BATT (yellow)	 Transmitting Receiving Ready to send Battery failure
Back-up battery Lithium 1/2 AA	 3.6 V/850 mAh
Current consumption from +9 V	typ. 140 mA
Power loss of module	typ. 1.2 W
Weight	approx. 500 g (1.1 lb.)



The CP 521 SI (Serial Interface) communications module is a powerful I/O module with its own central processor.

A separate manual is available for this module. It can be ordered under the order number 6ES5 998-1UD11.

The following is an overview of the module's mode of operation.

Function

The CP 521 SI can be used for unidirectional and bidirectional data exchange.

Unidirectional Data Exchange

For unidirectional data traffic, the CP 521 is provided with a printer driver. If the printer driver is used, the following must be connected to the serial interface of the CP 521 SI:

- A printer with TTY interface (active) or
- A printer with V.24 (RS 232C) interface.

This enables you to log process states and process disturbances. The output of messages on the printer does not extend the response time of the programmable controller.

The following messages and texts can be output:

- Message texts, which you have configured on a memory submodule in data blocks DB 2 to 63.
- Time of day and date, which are provided by the module's own clock
- Values for variables which are transmitted to the CP 521 SI via the I/O bus.

The message texts are stored on an EPROM or EEPROM memory submodule (up to 8/16 Kbytes).

Bidirectional Data Exchange

The following drivers are implemented for the bidirectional data exchange:

- ASCII driver, transparent
- ASCII driver, interpreting
- ”3964(R)” driver
- SINEC L1 driver, master (point-to-point)
- SINEC L1 driver, slave
- Terminal driver

The use of these drivers enables the transmission of data frames between the CPU and an I/O device connected to the CP 521 SI.

The maximum data flow rate is 6 bytes of useful data per 2 program cycles; i.e. at a program cycle time of, for example, 50 ms a maximum of 60 bytes per second can be transmitted.

The following terminals and communications devices can be used as I/O devices:

- Keyboard
- Terminal
- Another CP 521 SI
- CP 523
- S5-95U with second serial interface
- CP 524/CP 525-2 (in connection with special driver 6ES5 897-2AB11)
- CPU 944 (with ASCII driver, 3964(R) driver)
- Other I/O devices with serial interface, e.g. bar code readers

Which of the I/O devices and transmission modes are used depends on the intended application of data transmission. In the bidirectional data exchange mode of the module you are, for example, able to network programmable controllers (point-to-point link).

I/O devices and CP 521 SI are connected with each other via a serial interface. Either a passive TTY interface or a 24 V voltage interface is available (programmable).

Parameterizing (matching) of the I/O interface and configuring of the message texts are supported by the DB editor of programmers. The parameters of the I/O interface are stored either on a memory submodule in DB1 or are directly transmitted in the user program. The CP 521 SI can be programmed and operated wihtout the COM software.

Integrated Real-Time Clock

The CP 521 SI has it's own real-time clock which is battery-backed when the module is in the de-energized state. Independent of the type of function selected for the CP 521 SI, the clock data can be read from the CPU and can be used in the user program for date and time-dependent tasks.

Installation

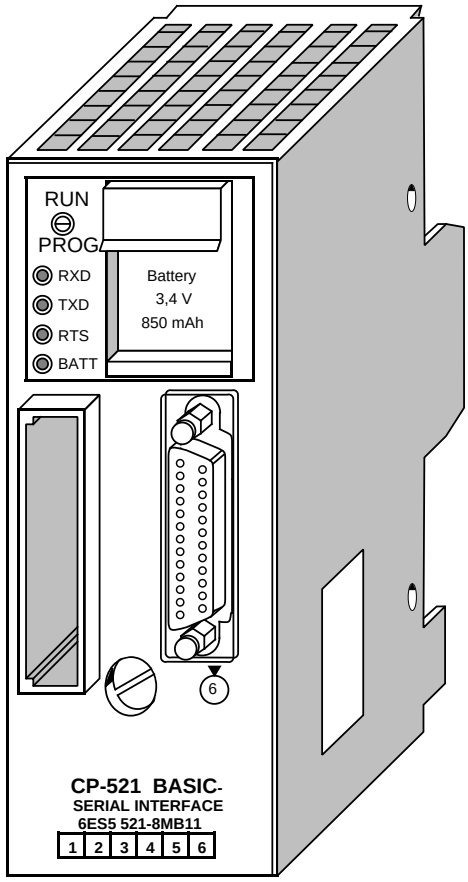
As other I/O modules, the CP 521 SI is mounted on the bus unit.

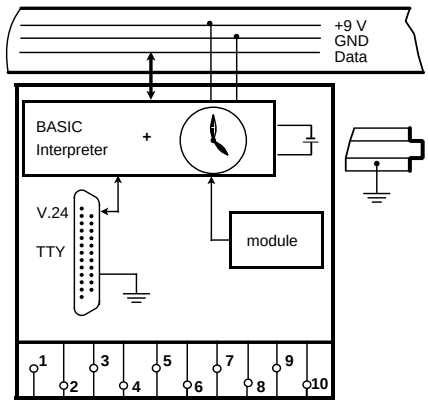
- With the S5-95F, a maximum of 7 CP 521 SI modules (limited by current consumption) can be plugged in only at slots 0 to 7.
- The module has no connection to the terminal block.

Adressing

The CP 521 SI is addressed like a 4-channel analog module.

CP 521 BASIC Communications Module (6ES5 521-8MB11)





Technical Specifications

Galvanic isolation		TTY signals are isolated
Serial interface		V.24 (RS-232-C)/TTY, passive (active)
Memory submodule		EPROM/EEPROM/ RAM
Real-time clock - accuracy tg		±1 s/day at 25 C (77 F)
- variation due to temperature change tA		-10 to +70 C
- (ambient temperature TV in C)		1 s to -11 s according to data sheet
Transmission mode		asynchronous 10-bit character frame /11-bit character frame
Baud rate		110 to 9600 Baud
LED displays: -TXD - RXD - RTS - BATT		 Send data Receive data Ready to send Battery low signal (yellow LED)
Permissible cable lengths -TTY, dependent on voltage drop on the + line: - typical for receiver - typical for sender - V.24		 1.5 V + 0.9 V 15 m (50 feet)
Back-up battery Lithium 1/2 AA		Battery failure display, yes 3.4 V/850 mAh
Life expectancy		1 year minimum
Degree of protection		IP 20
Current consumption from + 9 V (CPU)	typ.	180 mA
Power loss of the module	typ.	1.6 W
Weight	approx.	500 g (1 lb 5 oz.)

NOTE: It is only possible to run the CP 521 with the interrupt processing if the interrupts are disabled at the end of the OB1 cycle and enabled again at the beginning of the OB1 cycle.

The CP 521 BASIC communications processor is a powerful I/O module with its own central processor.

A separate manual for this module is available. The order number is 6ES5 998-0UW11.

A brief overview of the functions of this module follows.

Function

This module comes with a special COM software package that is required for generating and storing BASIC programs (on a floppy disk or an EPROM submodule).

Since the CP 521 includes a basic interpreter, you can create and run BASIC programs that exchange data with a CPU and a connected peripheral device. Use a programmer or a PC terminal and the COM software to program the BASIC interpreter.
You can store the BASIC programs in the module's own battery backed-up RAM or on a memory submodule that can be plugged in.

Connect programmers or PC terminals to the CP 521 via a serial interface. You can choose (by setting parameters) between a passive TTY current-loop interface or a RS-232 C V.24 interface to connect a programmer or terminal. Connect a printer to the unidirectional V.24 interface of the module to print listings or messages.

Change parameter settings for the peripheral interface by using a BASIC command or by using the BASIC program.

The CP 521 has an integral real-time clock that can be backed up by a battery. You can use the clock data in unidirectional data traffic to log process statuses or process malfunctions.

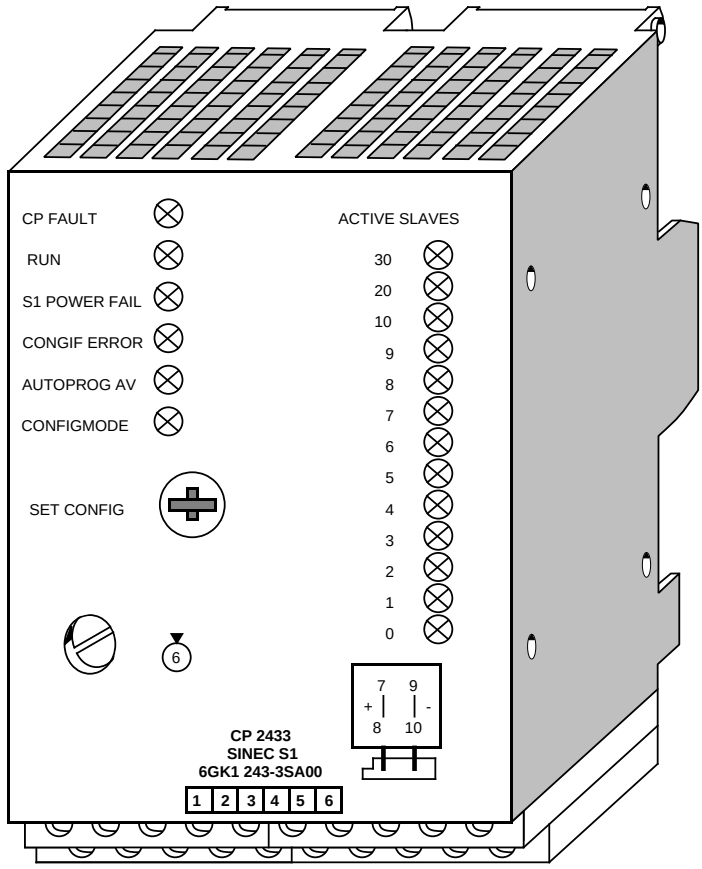
Installation

- Install the CP 521 BASIC communications module on the bus module like any other I/O module (see section 5)
- With the S5-95F, a maximum of 5 communications modules (limited by current consumption) can be plugged in only at slots 0 to 7.
- The module has no connection to the terminal block.
- Connect the printer to the module via a 25-pin sub-D female connector.

Addressing

The module is addressed like a 4-channel analog module.

Master Module for Actuator-Sensor-Interface CP 2433 (6GK1 243-3SA00)

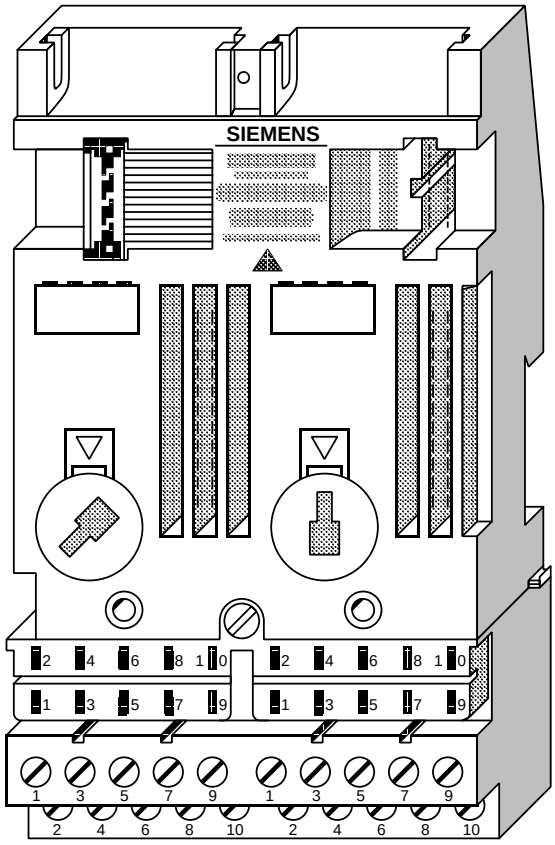


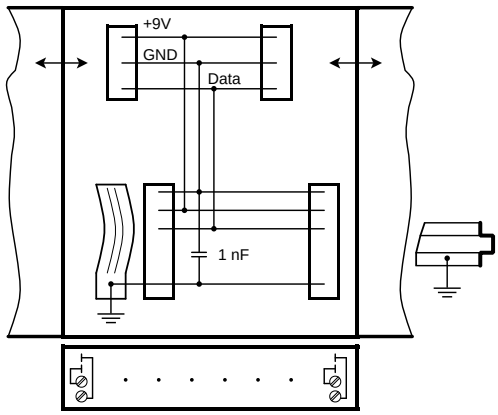
Technical Specifications

Interface:	AS Interface	Required slots:	2
		Permissible slots:	0 and 2, 4 and 6 1 and 3, 5 and 7
Configuring:	via switch or function blocks		
		Addressing:	16 bytes in the analog area
AS-i profiles:	M0 without function blocks		
		Current consumption:	
	M1 with function blocks	- 9 V DC from bus unit	200 mA
		- 31 V DC from AS-i cable	100 mA
Number of AS-i slaves:	max. 31	Weight:	300g (10.6 oz.)
AS-i bus cycle time:	max. 5 ms		

A.2.7 Bus Units

Bus Unit (SIGUT) (6ES5 700-8MA11)



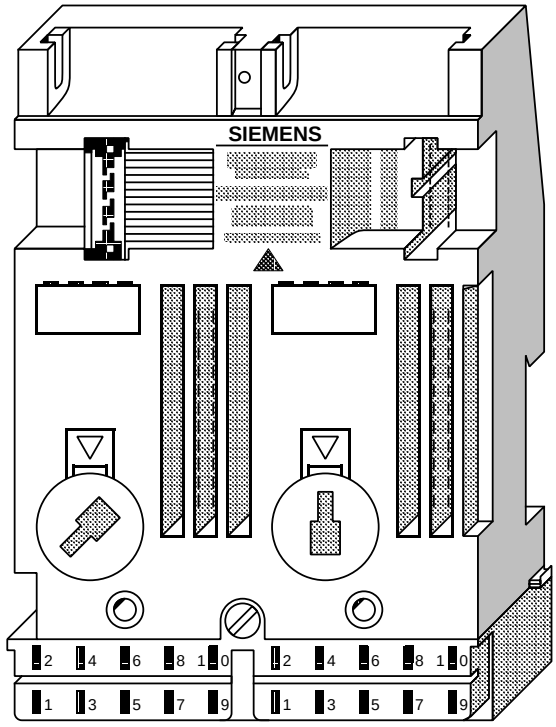


Technical specifications

Type of connection		SIGUT screw-type terminals
Number of plug-in modules		2
Number of bus units per programmable controller	max.	16
Connection between two bus units		flat ribbon
Number of terminals		10 per slot
Rated insulation voltage (+9 V to ⏚)		12 V AC
- insulation group		1×B
- tested with		500 V AC
Conductor cross sectional area		
- stranded*		2×0.5 to 1.5 mm^2
- solid		2×0.5 to 2.5 mm^2
Current consumption - from +9 V (CPU)	typ.	1 mA
Dimensions W×H×D (in mm)		91.5×162×39
Weight	approx.	300 g (10.6 oz.)

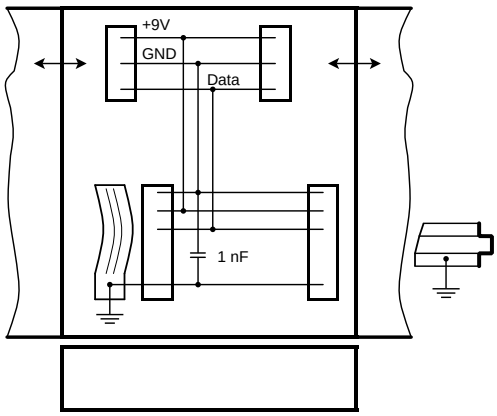
* with core end sleeves

Bus Unit (Crimp-Snap-In Connections) (6ES5 700-8MA21)



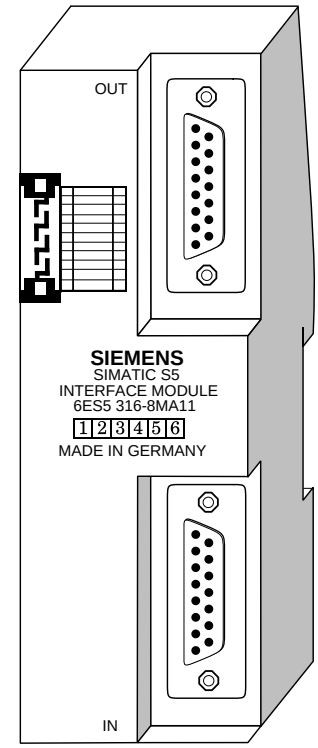
Technical Specifications

Type of connection		Crimp snap-in
Number of plug-in modules		2
Number of bus units per programmable controller	max.	16
Connection between two bus units		flat ribbon
Number of terminals		10 per slot
Conductor cross sectional area - stranded		0.5 to 1.5 mm^2
Rated insulation voltage (+9 V to ⏚)		12 V AC
- insulation group		1×B
- tested with		500 V AC
Current consumption - from +9 V (CPU)	typ.	1 mA
Dimensions W×H×D (in mm)		91.5×135×39
(in in.)		3.6×5.3×1.5
Weight	approx.	250 g (8.8 oz.)



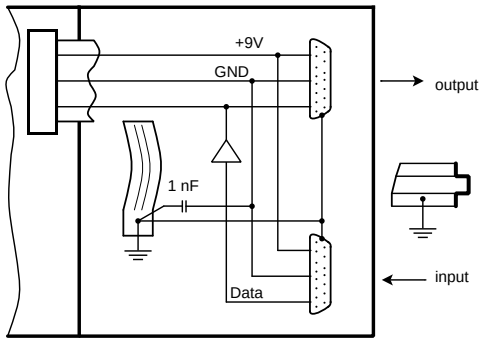
A.2.8 Interface Modules

IM 316 Interface Module (6ES5 316-8MA12)



Technical specifications

Current supply to the expansion unit	max.	2.5 A
Number of interface modules per PLC	max.	4
Cable connectors for the IM 316		
- Cable connector (0.5 m/1.6 ft.)		6ES5 712-8AF00
- Cable connector (2.5 m/8.2 ft.)		6ES5 712-8BC50
- Cable connector (.0 m/16.4 ft.)		6ES5 712-8BF00
- Cable connector (10 m/33 ft.)		6ES5 712-8CB00
Cable insulation in ducts		permissible
Permissible potential difference between ⏚ (IM 316) and central ground point (CPU)		±1 V
Rated insulation voltage (+9 V to ⏚)		12 V AC
- insulation group		1×B
Dimensions W×H×D in mm		45.4×135×39
Current consumption - from +9 V (CPU)	typ.	27 mA
Weight	approx.	120 g (4.2 oz.)



B Dimension Drawings

B.1 S5-95F Programmable Controller . . . . . B - 1

B.2 Bus Units . . . . . B - 2

B.3 Interface Modules . . . . . B - 4

B.4 Standard Mounting Rails . . . . . B - 5

Figures

B-1	Dimension Drawing of the S5-95F	B - 1
B-2	Dimension Drawing of the Bus Unit (Crimp Snap-in Connections) with I/O Module	B - 2
B-3	Dimension Drawing of the Bus Unit (SIGUT Screw-Type Terminals) with I/O Module	B - 3
B-4	Dimension Drawing of the IM 316 Interface Module (6ES5 316-8MA12) . . .	B - 4
B-5	Cross Sections of Standard Mounting Rails	B - 5
B-6	Dimension Drawing of the 483-mm (19-in.) Standard Mounting Rail	B - 5
B-7	Dimension Drawing of the 530-mm (20.9-in.) Standard Mounting Rail	B - 6
B-8	Dimension Drawing of the 830-mm (32.7-in.) Standard Mounting Rail	B - 6
B-9	Dimension Drawing of the 2-m (6.6-ft.) Standard Mounting Rail	B - 6

B Dimension Drawings

B.1 S5-95F Programmable Controller

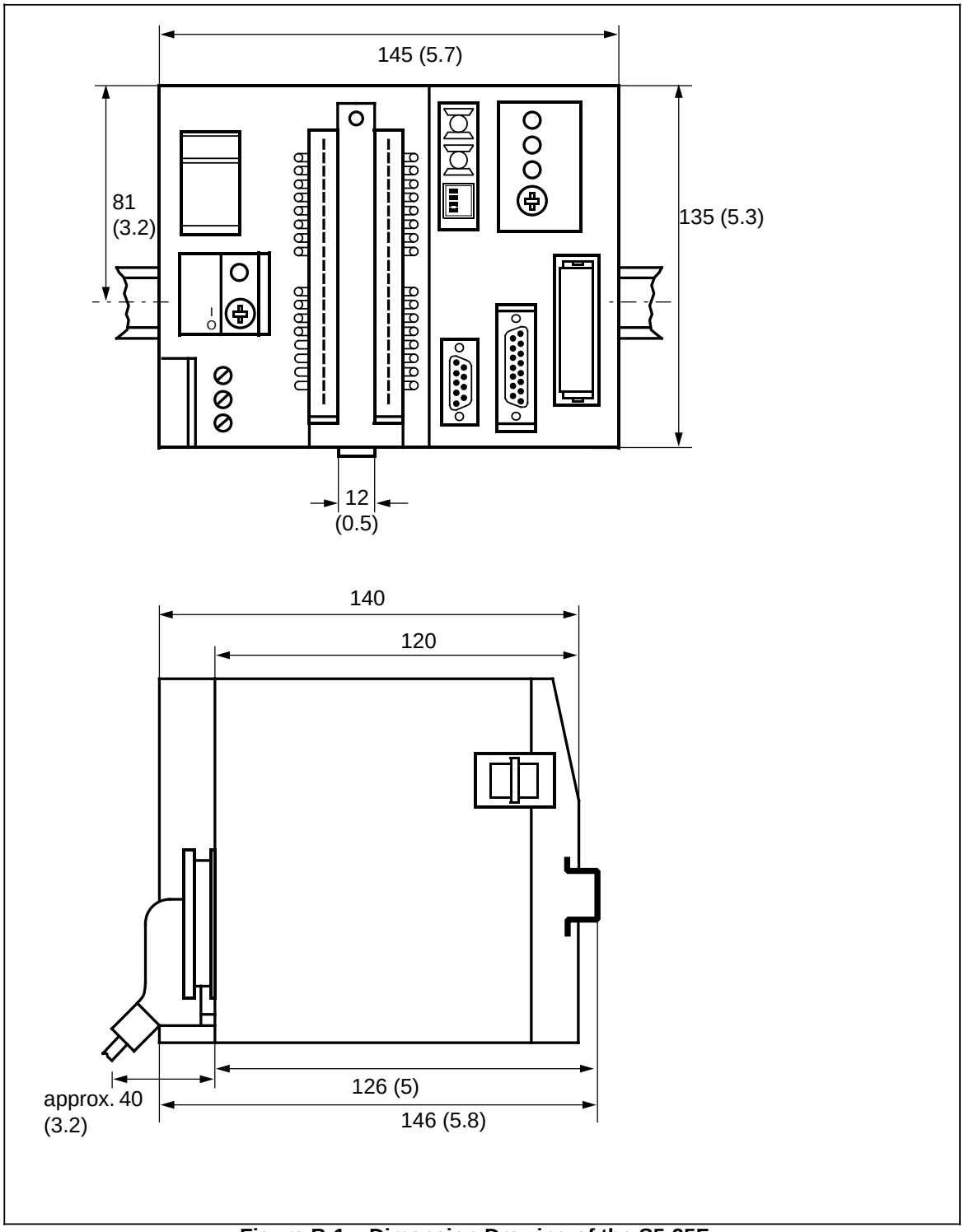


Figure B-1. Dimension Drawing of the S5-95F

B.2 Bus Units

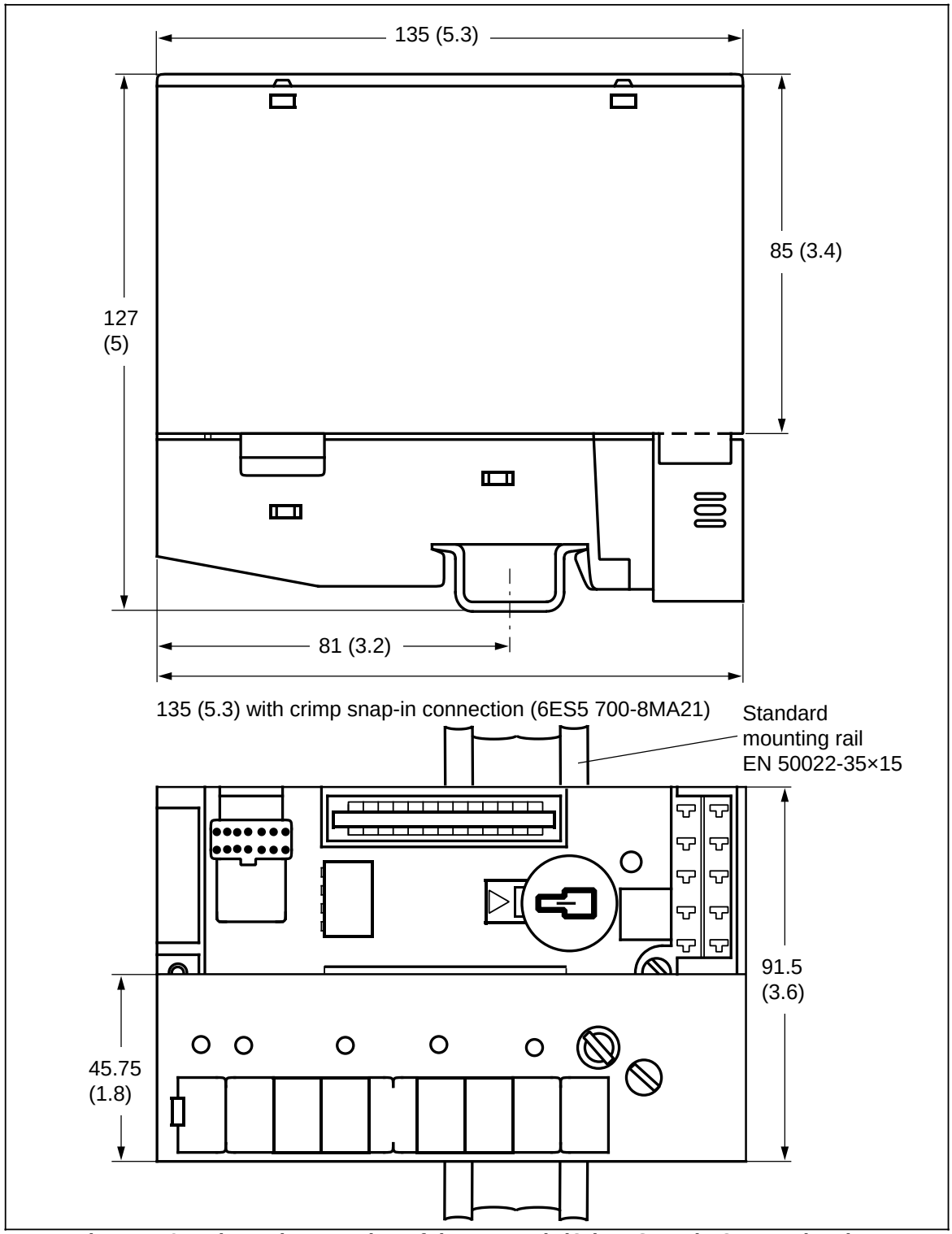


Figure B-2. Dimension Drawing of the Bus Unit (Crimp Snap-in Connections) with I/O Module

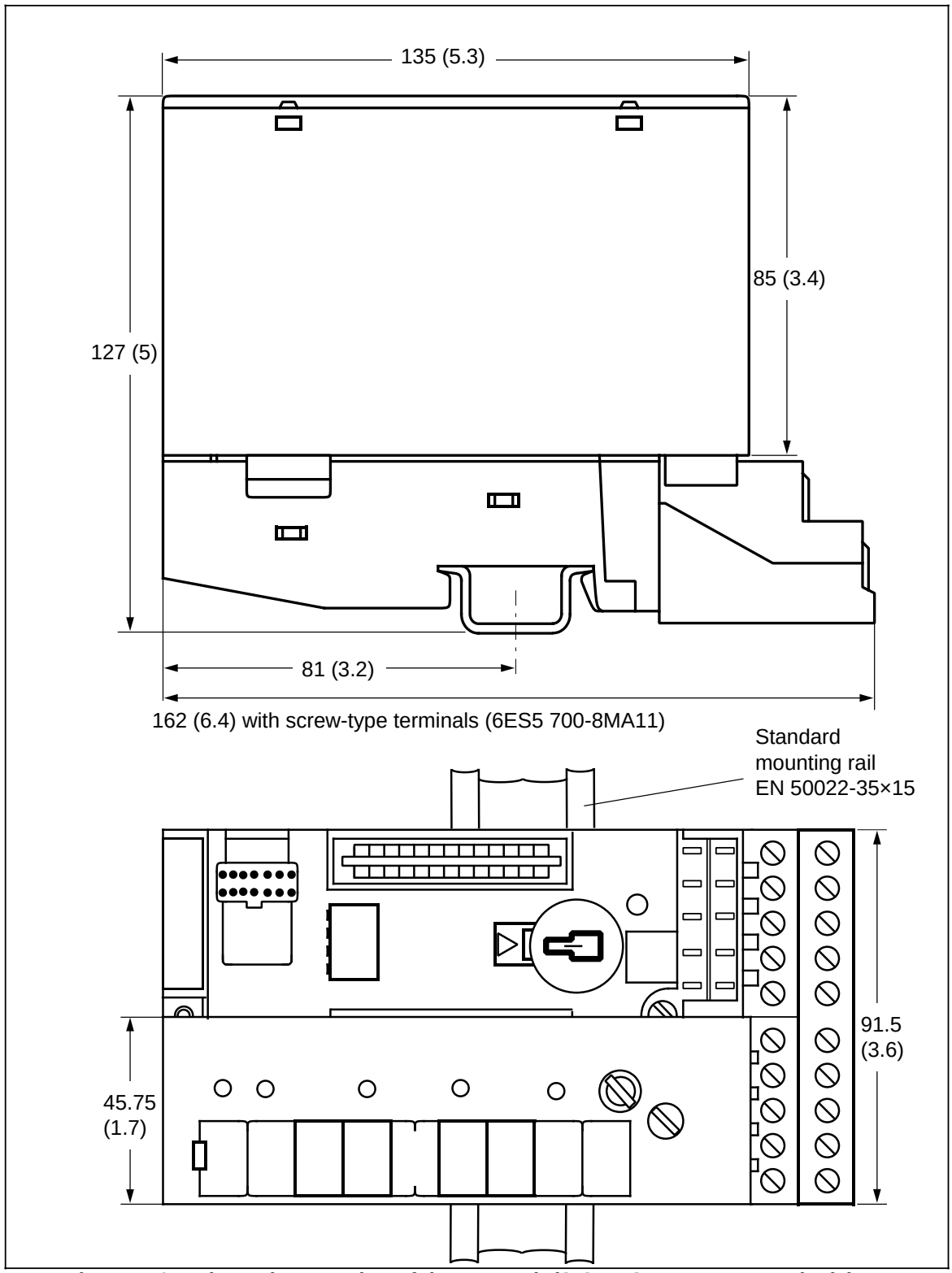


Figure B-3. Dimension Drawing of the Bus Unit (SIGUT Screw-type Terminals) with I/O Module

B.3 Interface Modules

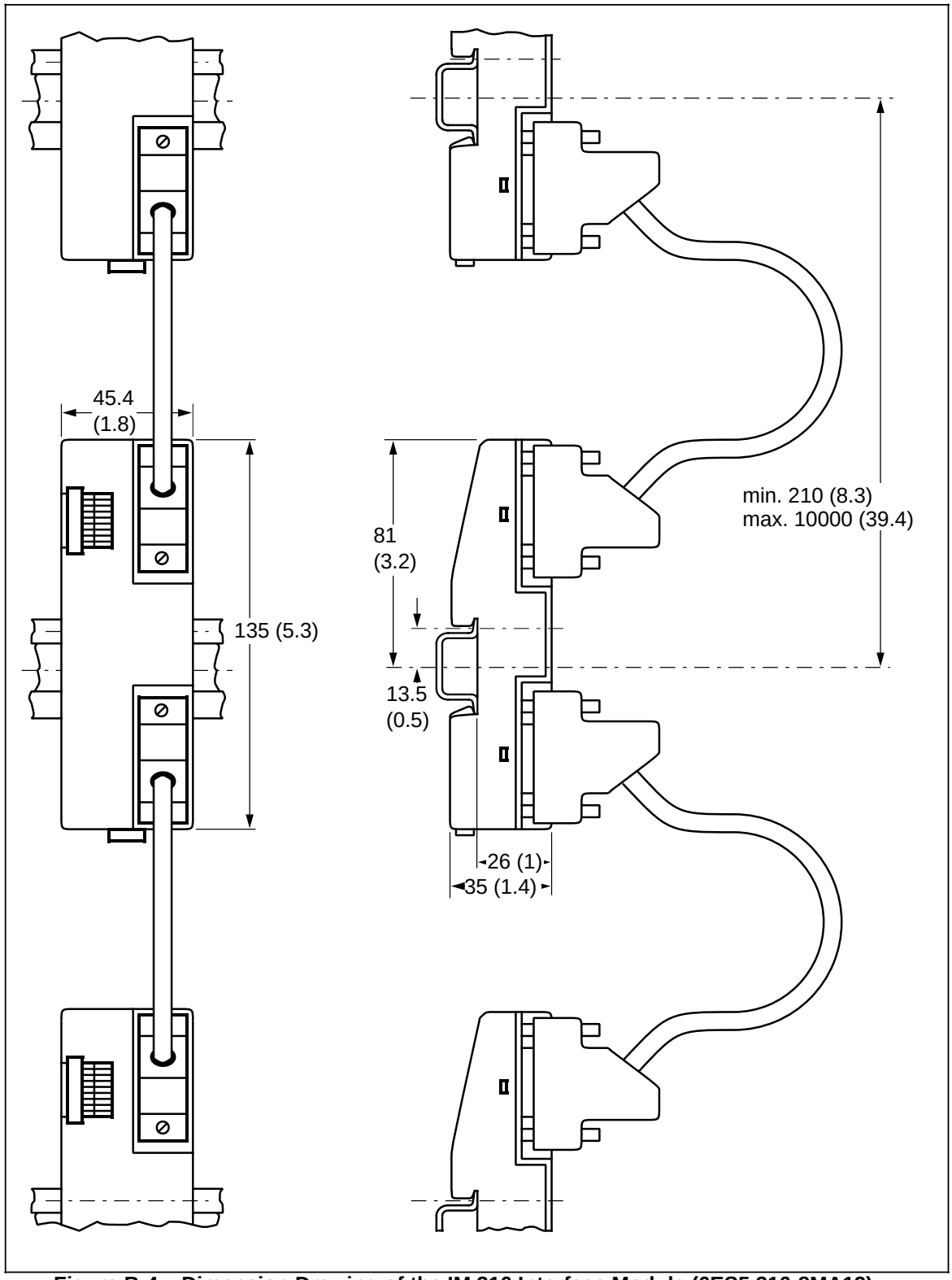


Figure B-4. Dimension Drawing of the IM 316 Interface Module (6ES5 316-8MA12)

B.4 Standard Mounting Rails

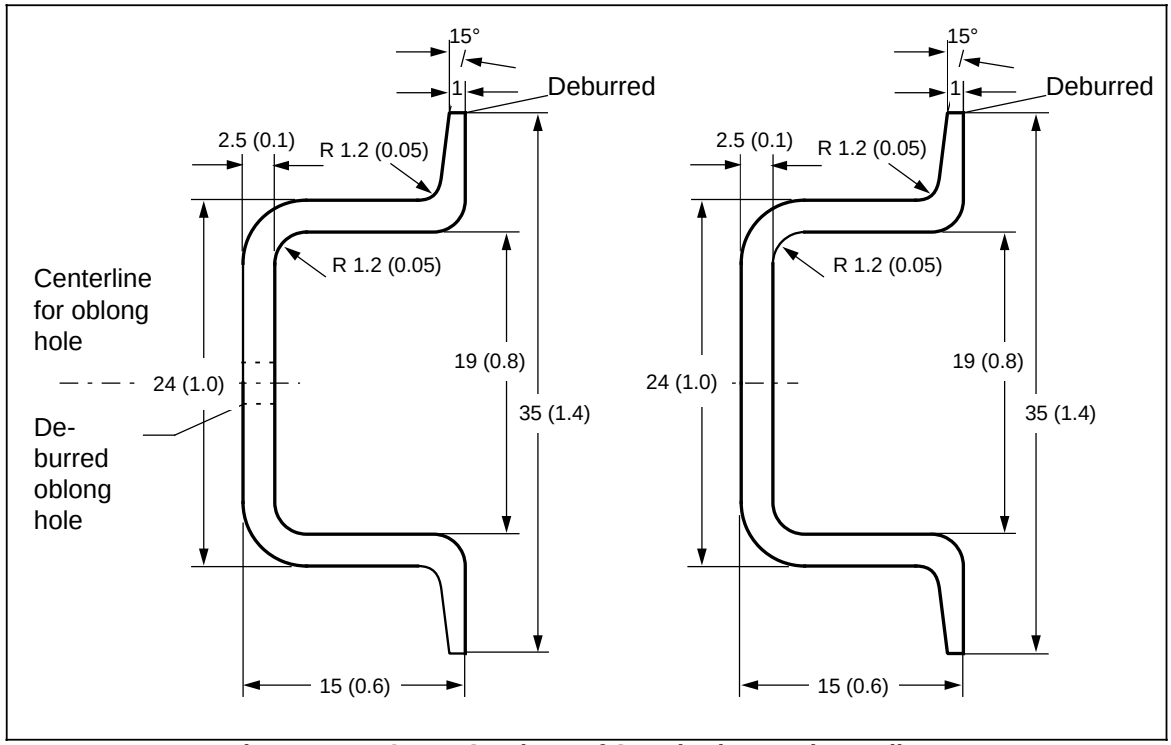


Figure B-5. Cross Sections of Standard Mounting Rails

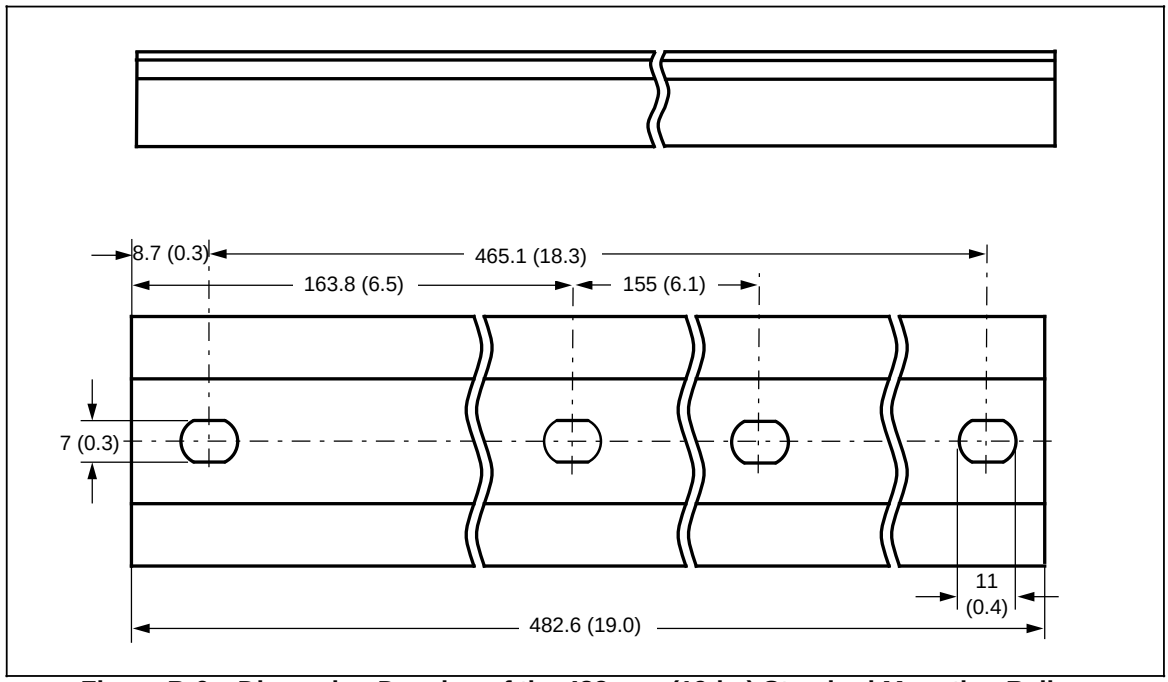


Figure B-6. Dimension Drawing of the 483-mm (19-in.) Standard Mounting Rail

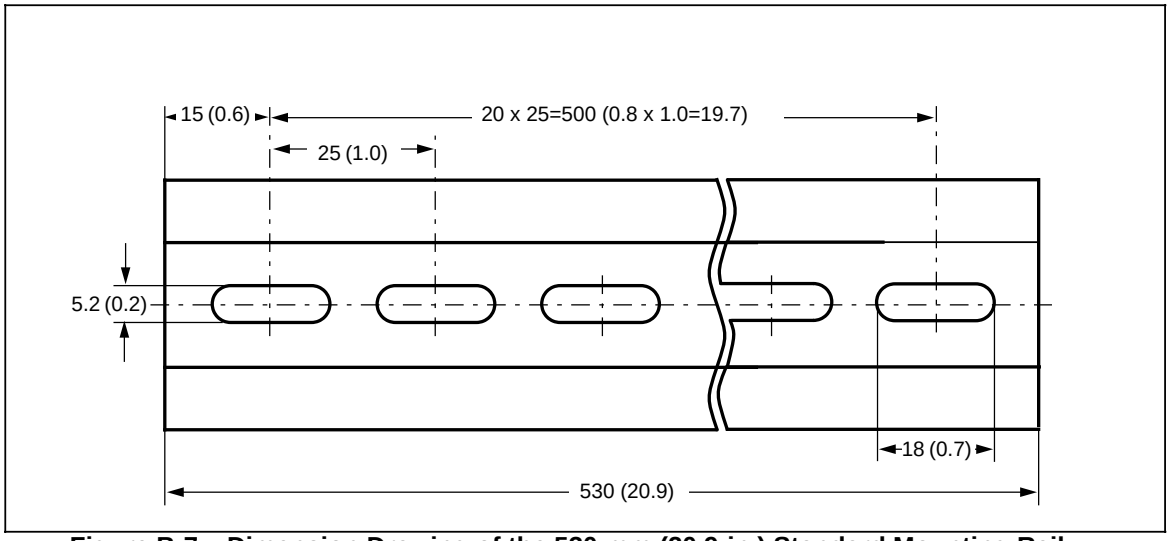


Figure B-7. Dimension Drawing of the 530-mm (20.9-in.) Standard Mounting Rail

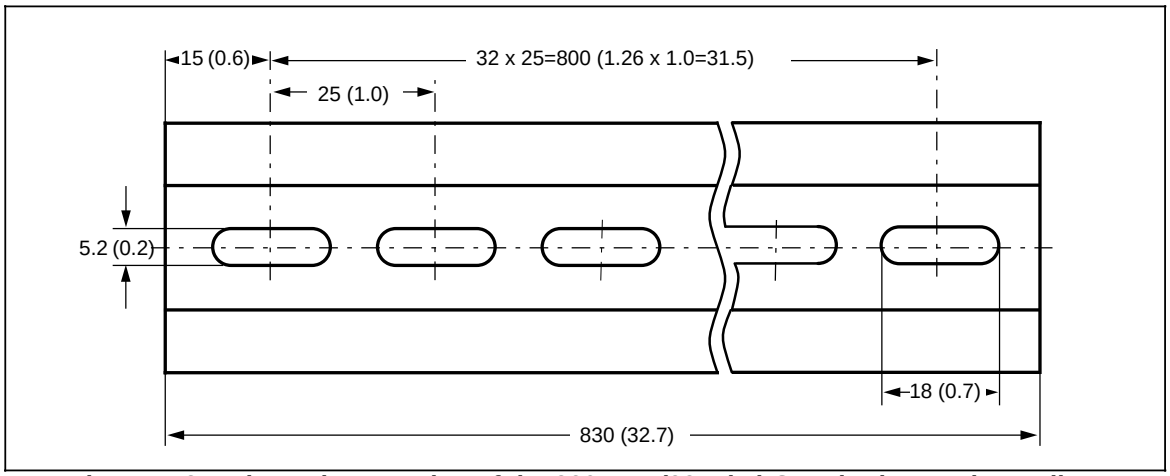


Figure B-8. Dimension Drawing of the 830-mm (32.7-in.) Standard Mounting Rail

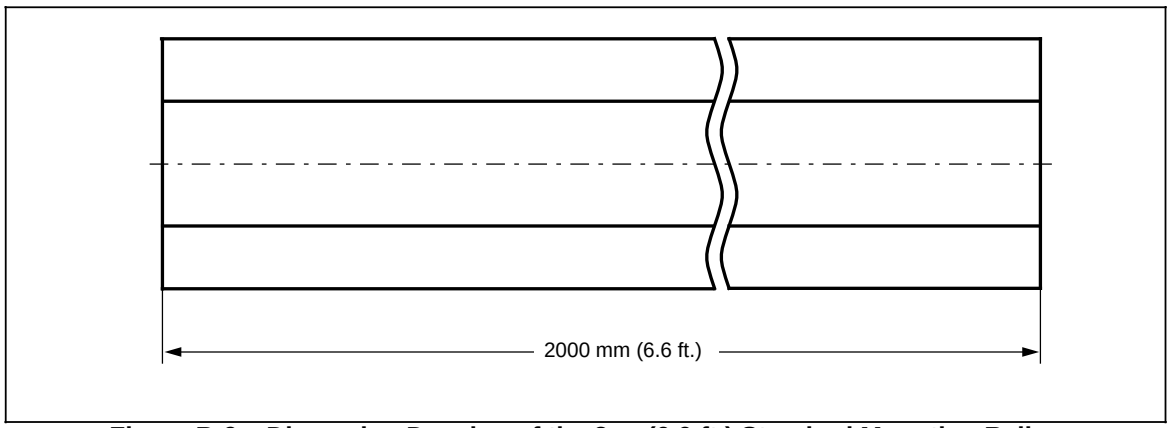


Figure B-9. Dimension Drawing of the 2-m (6.6-ft.) Standard Mounting Rail

C Operations List

C.1	Operations List	C - 1
C.1.1	Basic Operations	C - 1
C.1.2	Supplementary Operations	C - 7
C.1.3	System Operations	C - 12
C.1.4	Evaluation of CC 1 and CC 0	C - 12
C.2	Machine Code Listing	C - 13
C.3	List of Abbreviations	C - 16

C Operations List

C.1 Operations List

The following tables list the typical execution times for the individual operations. If you wish to calculate the runtime of a program part, then you must add synchronization times to the typical execution time. After an execution time of approx. 5 ms each, the S5-95F compiler initiates a synchronization call for the synchronization of the two subunits. The synchronization calls result in about a 10% system onload. The system load increases even more when programmer, SINEC L1, OB 2, OB 3 and/or OB 13 are involved (see section 7.4.2).

C.1.1 Basic Operations

for organization blocks (OB) for function blocks (FB)
for program blocks (PB) for sequence blocks (SB)

Operation (STL)	Permissible Operands	1 RLO-depend. 2 RLO-affected 3 RLO reloaded			Typical Execution Time in µs		Function
		1	2	3	On-board	Ext. I/O	
Boolean Logic Operations							
A	I/Q	N	Y	N	0.8	2.4	Scan operand for "1" and combine with RLO through logic AND.
	F	N	Y	N	2.4		
	T0 to 63 T64 to 127	N	Y	N	180 15		
	C	N	Y	N	2.4		
AN	I, Q	N	Y	N	0.8	2.4	Scan operand for "0" and combine with RLO through logic AND.
	F	N	Y	N	2.4		
	T0 to 63 T64 to 127	N	Y	N	180 10		
	C	N	Y	N	2.4		
O	I, Q	N	Y	N	0.8	2.4	Scan operand for "1" and combine with RLO through logic OR.
	F	N	Y	N	2.4		
	T0 to 63 T64 to 127	N	Y	N	180 15		
	C	N	Y	N	2.4		
ON	I, Q	N	Y	N	0.8	2.4	Scan operand for "0" and combine with RLO through logic OR.
	F	N	Y	N	2.4		
	T0 to 63 T64 to 127	N	Y	N	180 15		
	C	N	Y	N	2.4		
O		N	Y	Y	0.8		Combine AND operations through logic OR.
A(		N	Y	Y	3.6		Combine expressions enclosed in parentheses through logic AND (6 nesting levels).
O(		N	Y	Y	2		Combine expressions enclosed in parentheses through logic OR (6 nesting levels).
)		N	Y	Y	2		Close parentheses (conclusion of a parenthetical expression).

for organization blocks (OB) for function blocks (FB)
for program blocks (PB) for sequence blocks (SB)

Operation (STL)	Permissible Operands	1 RLO depend. 2 RLO affected 3 RLO reloaded			Typical Execution time in µs		Function
		1	2	3	On-board	Ext. I/O	
Set / Reset Operations							
S	I, Q	Y	N	Y	1.2	3.6	Set operand to "1".
	F	Y	N	Y	3.6		
R	I, Q	Y	N	Y	1.2	3.6	Reset operand to "0".
	F	Y	N	Y	3.6		
=	I, Q	N	N	Y	0.8	3.2	Assign value of RLO to operand.
	F	N	N	Y	3.2		
Load Operations							
L	IB	N	N	N	2.4	5.2	Load an input byte from the PII into ACCU 1.
L	QB	N	N	N	2.4	5.2	Load an output byte from the PIQ into ACCU 1.
L	IW	N	N	N	2.4	6	Load an input word from the PII into ACCU 1: byte n ACCU 1 (bits 8-15); byte n+1 ACCU 1 (bits 0-7)
L	QW	N	N	N	3.6	6	Load an output word from the PIQ into ACCU 1: byte n ACCU 1 (bits 8-15); byte n+1 ACCU 1 (bits 0-7)
L	PY0 to 31 PY64 to 127	N N	N N	N N	30		Permissible only in OB13. Load an input byte of the external digital/analog inputs from the interrupt PII into ACCU 1.
	PY32/33 PY35 PY36 to 38 PY59	N	N	N	70 to 90		Load an input byte of the onboard I/O inputs into ACCU 1.
L	PW 0 to 30 PW 64 to 126	N	N	N	32		Permissible only in OB13. Load an input word of the external digital/analog inputs from the interrupt PII into ACCU 1.
	PW 32 PW 36 to 38	N	N	N	120 to 150		Load an input word of the onboard I/O inputs into ACCU 1.
L	FY	N	N	N	5.2		Load a flag byte into ACCU 1.
L	FW	N	N	N	6		Load a flag word into ACCU 1: byte n ACCU 1 (bits 8-15); byte n+1 ACCU 1 (bits 0-7).
L	DL	N	N	N	15.2		Load a data word (left-hand byte) of the current data block into ACCU 1.
L	DR	N	N	N	16		Load a data word (right-hand byte) of the current data block into ACCU 1.

for organization blocks (OB) for function blocks (FB)
for program blocks (PB) for sequence blocks (SB)

Operation (STL)	Permissible Operands	1 RLO depend. 2 RLO affected 3 RLO reloaded			Typical Execution Time in µs		Function
		1	2	3	On-board	Ext. I/O	
Load Operations (cont.)							
L	DW	N	N	N	16.4		Load a data word of the current data block into ACCU 1: byte n ACCU 1 (bits 8-15); byte n+1 ACCU 1 (bits 0-7).
L	KB	N	N	N	2.4		Load a constant (1-byte number) into ACCU 1.
L	KS	N	N	N	2.4		Load a constant (2 characters in ASCII format) into ACCU 1.
L	KF	N	N	N	2.4		Load a constant (fixed-point number) into ACCU 1.
L	KH	N	N	N	2.4		Load a constant (hexadecimal code) into ACCU 1.
L	KM	N	N	N	2.4		Load a constant (bit pattern) into ACCU 1.
L	KY	N	N	N	2.4		Load a constant (2-byte number) into ACCU 1.
L	KT	N	N	N	2.4		Load a constant (time in BCD) into ACCU 1.
L	KC	N	N	N	2.4		Load a constant (count in BCD) into ACCU 1.
L	T0 to 63 T64 to 127	N	N	N	190 18		Load a time or count (in binary code) into ACCU 1.
	C	N	N	N	6.8		
LC	T0 to 63 T64 to 127	N	N	N	210 39		Load times or counts (in BCD) into ACCU 1.
	C	N	N	N	26		
Transfer Operations							
T	IB	N	N	N	0.8	2	Transfer the contents of ACCU 1 to an input byte (into the PII).
T	QB	N	N	N	0.8	2	Transfer the contents of ACCU 1 to an output byte (into the PIQ).
T	IW	N	N	N	2.8	5.6	Transfer the contents of ACCU 1 to an input word (into the PII): ACCU 1 (bits 8-15) byte n; ACCU 1 (bits 0-7) byte n+1.
T	QW	N	N	N	2.8	5.6	Transfer the contents of ACCU 1 to an input word (into the PII): ACCU 1 (bits 8-15) byte n; ACCU 1 (bits 0-7) byte n+1.

for organization blocks (OB) for function blocks (FB)
for program blocks (PB) for sequence blocks (SB)

Operation (STL)	Permissible Operands	1 RLO depend. 2 RLO affected 3 RLO reloaded			Typical Execution Time in µs		Function
		1	2	3	On-board	Ext. I/O	
Transfer Operations (cont.)							
T	PY0 to 31 PY64 to 127	N	N	N	45		Permissible only in OB13. Transfer the contents of ACCU 1 to the interrupt PIQ with updating of the PIQ.
	PY32/33 PY34	N	N	N	45 25		Transfer the contents of ACCU 1 to the output with updating of the PIQ.
T	PW0 to 30 PW64 to 126	N	N	N	35		Permissible only in OB13. Transfer the contents of ACCU 1 to the interrupt PIQ with updating of the PIQ.
	PW32 PW33	N	N	N	60 40		Transfer the contents of ACCU 1 to the output with updating of the PIQ.
	PW36 to 38	N	N	N	210		Reset counter to "0". Transfer comparison value.
T	FY	N	N	N	2		Transfer the contents of ACCU 1 to a flag byte.
T	FW	N	N	N	5.6		Transfer the contents of ACCU 1 to a flag word (into the PIQ): ACCU 1 (bits 8-15) byte n; ACCU 1 (bits 0-7) byte n+1.
T	DL	N	N	N	12.8		Transfer the contents of ACCU 1 to a data word (left-hand byte).
T	DR	N	N	N	13.6		Transfer the contents of ACCU 1 to a data word (right-hand byte)
T	DW	N	N	N	15.6		Transfer the contents of ACCU 1 to a data word.
Time Operations							
SP	T	Y	N	Y	180		Start a timer (stored in ACCU 1) as a signal-contracting pulse on the leading edge of the RLO.
SE	T	Y	N	Y	180		Start a timer (stored in ACCU 1) as extended pulse (signal contracting and stretching) on the leading edge of the RLO.
SI	T	Y	N	Y	180		Start an on-delay timer (stored in ACCU 1) on the leading edge of the RLO.
SS	T	Y	N	Y	180		Start a stored on-delay timer (stored in ACCU 1) on the leading edge of the RLO.
SF	T	Y	N	Y	180		Start an off-delay timer (stored in ACCU 1) on the trailing edge of the RLO.
R	T	Y	N	Y	5.6 to 8.4		Reset a timer.

for organization blocks (OB) for function blocks (FB)
for program blocks (PB) for sequence blocks (SB)

Operation (STL)	Permissible Operands	1 RLO depend. 2 RLO affected 3 RLO reloaded			Typical Execution Time in µs		Function
		1	2	3	On-board	Ext. I/O	
Counter Operations							
CU	C	Y	N	Y	14.4		Counter counts up 1
CD	C	Y	N	Y	14.8		Counter counts down 1
S	C	Y	N	Y	29.2		Set counter
R	C	Y	N	Y	7.2		Reset counter
Arithmetic Operations							
+F		N	N	N	10		Add two fixed-point numbers: ACCU 1+ACCU 2. CC 1 / CC 0 / OV are affected.
-F		N	N	N	8.8		Subtract one fixed-point number from another: ACCU 2 - ACCU 1. CC 1 / CC 0 / OV are affected.
Comparison Operations							
!=F		N	Y	N	11.2		Compare two fixed-point numbers for "equal to": If ACCU 2=ACCU 1, the RLO is "1". CC 1 / CC 0 are affected.
><F		N	Y	N	10.8		Compare two fixed-point numbers for "not equal to": If ACCU 2 ACCU 1, the RLO is "1". CC 1 / CC 0 are affected.
>F		N	Y	N	10.8		Compare two fixed-point numbers for "greater than": If ACCU 2 > ACCU 1, the RLO is "1". CC 1 / CC 0 are affected.
>=F		N	Y	N	10.8		Compare two fixed-point numbers for "greater than or equal to": If ACCU 2 ACCU 1, the RLO is "1". CC 1 / CC 0 are affected.
<F		N	Y	N	10.8		Compare two fixed-point numbers for "less than": If ACCU 2 < ACCU 1, the RLO is "1". CC 1 / CC 0 are affected.
<=F		N	Y	N	10.8		Compare two fixed-point numbers for "less than or equal to": If ACCU 2 ACCU 1, the RLO is "1". CC 1 / CC 0 are affected.

for organization blocks (OB) for function blocks (FB)
for program blocks (PB) for sequence blocks (SB)

Operation (STL)	Permissible Operands	1 RLO depend. 2 RLO affected 3 RLO reloaded			Typical Execution Time in µs		Function
		1	2	3	On-board	Ext. I/O	
Block Call Operations							
JU	PB	N	N	Y	56		Jump unconditionally to a program block.
JU	FB	N	N	Y	56		Jump unconditionally to a function block.
JU	SB	N	N	Y	56		Jump unconditionally to a sequence block.
JC	PB	Y	Y[1)]	Y	56		Jump conditionally to a program block.
JC	FB	Y	Y[1)]	Y	56		Jump conditionally to a function block.
JC	SB	Y	Y[1)]	Y	56		Jump conditionally to a sequence block.
C	DB	N	N	N	16		Call a data block.
G	DB	N	N	Y	59		Generate or delete a data block.
Return Operations							
BE		N	N	Y	36		Block end (termination of a block)
BEC		Y	Y[1)]	Y	36		Block end, conditional
BEU		N	N	Y	36		Block end, unconditional (BEU cannot be used in organization blocks.)
"**No**" **Operations**							
NOP 0		N	N	N	-		No operation (all bits reset)
NOP 1		N	N	N	-		No operation (all bits set)
Stop Operation							
STP		N	N	N	0.4		Stop: scanning is still completed before a stop. Error ID ”STS” is set in the ISTACK.
Display Generation Operations							
BLD 130		N	N	N	-		Display generation operation for the programmer: carriage return generates blank line.
BLD 131		N	N	N	-		Display generation operation for the programmer: switch to statement list (STL).
BLD 132		N	N	N	-		Display generation operation for the programmer: switch to control system flowchart (CSF).
BLD 133		N	N	N	-		Display generation operation for the programmer: switch to ladder diagram (LAD).
BLD 255		N	N	N	-		Display generation operation for the programmer: terminate a segment.

1) RLO is set to "1"

C.1.2 Supplementary Operations

for organization blocks (OB) for function blocks (FB)
for program blocks (PB) for sequence blocks (SB)

Operation (STL)	Permisible Operands	1 RLO depend. 2 RLO affected 3 RLO reloaded			Typical Execution Time in µs		Function
		1	2	3	On-board	Ext. I/O	
Boolean Logic Operations							
A=	Formal operand	N	Y	N			AND operation: scan formal operand for "1". (Data type: BI)
	I/Q				44		
	F				30		
	T0 ... 63				200		
	T64 ... 127				40		
	C				24		
AN=	Formal operand	N	Y	N			AND operation: scan formal operand for "0". (Data type: BI)
	I/Q				44		
	F				30		
	T0 ... 63				200		
	T64 ... 127				40		
	C				24		
O=	Formal operand	N	Y	N			OR operation: scan formal operand for "1". (Data type: BI)
	I/Q				44		
	F,				30		
	T0 ... 63				200		
	T64 ... 127				40		
	C				24		
ON=	Formaloperand	N	Y	N			OR operation: scan formal operand for "0". (Data type: BI)
	I/Q				44		
	F				30		
	T0 ... 63				200		
	T64 ... 127				40		
	C				24		
AW		N	N	N	6.8		Combine contents of ACCU 2 and ACCU 1 through logic AND (word operation). Result is stored in ACCU 1. CC 1 / CC 0 are affected.
OW		N	N	N	6.8		Combine contents of ACCU 2 and ACCU 1 through logic OR (word operation). Result is stored in ACCU 1. CC 1 / CC 0 are affected.
XOW		N	N	N	6.8		Combine contents of ACCU 2 and ACCU 1 through logic EXCLUSIVE OR (word operation). Result is stored in ACCU 1. CC 1 / CC 0 are affected.

for organization blocks (OB) for function blocks (FB)
for program blocks (PB) for sequence blocks (SB)

Operation (STL)	Permissible Operands	1 RLO depend. 2 RLO affected 3 RLO reloaded			Typical Execution Time in µs		Function
		1	2	3	On-board	Ext. I/O	
Bit Operations							
TB	T0 to 63 T64 to 127	N	Y	N	180 13.2		Test a bit of a timer or counter word for "1".
	C				2		
TB	D	N	Y	N	15.6		Test a bit of a data word for "1".
TB	RS	N	Y	N	2		Test a bit of a data word in the system data area for "1".
TBN	T0 to 63 T64 to 127	N	Y	N	180 13.6		Test a bit of a timer or counter word for "0".
	C				2.4		
TBN	D	N	Y	N	16		Test a bit of a data word for "0".
TBN	RS	N	Y	N	2.4		Test a bit of a data word in the system data area or "0".
SU	T, C	N	N	Y	2.8		Set a bit of a timer or counter word unconditionally.
SU	D	N	N	Y	16.4		Set a bit of a data word unconditionally.
RU	T, C	N	N	Y	2.8		Reset a bit of a timer or counter word unconditionally.
RU	D	N	N	Y	16.4		Reset a bit of a data word unconditionally.
Set / Reset Operations							
S=	Formal operand I/Q	Y	N	Y	65		Set a formal operand (when RLO=1). (Data type: BI)
	F				36		
RB=	Formal operand I/Q	Y	N	Y	65		Reset a formal operand (when RLO=1). (Data type: BI)
	F				36		
RD=	Formal operand T	Y	N	Y	31		Reset a formal operand (digital) (when RLO=1).
	C				29		
==	Formal operand I/Q	Y	N	Y	64		Assign the value of the RLO to the status of the formal operand. (Data type: BI)
	F				35		
Timer and Counter Operations							
FR	T	Y	N	Y	7.6		Enable a timer/counter for cold restart. If RLO="1", - "FR T" restarts the timer - "FR C" sets, decrements, or increments the counter.
	C	Y	N	Y	7.6		

for organization blocks (OB) for function blocks (FB)
for program blocks (PB) for sequence blocks (SB)

Operation (STL)	Permissible Operands	1 RLO depend. 2 RLO affected 3 RLO reloaded			Typical Execution Time in µs		Function
		1	2	3	On-board	Ext. I/O	
Timer and Counter Operations (cont.)							
FR=	Formal operand T	Y	N	Y	28		Enable formal operand (timer/ counter) for cold restart (for detailed description, see "FR" operation).
	C	Y	N	Y	27		
SP=	Formal operand T0 to 127	Y	N	Y	200		Start a timer (formal operand) as pulse with the value stored in ACCU 1.
SD=	Formal operand T0 to 127	Y	N	Y	200		Start an on-delay timer (formal operand) with the value stored in ACCU 1.
SEC=	Formal operand T0 to 127	Y	N	Y	200		Start a timer (formal operand) as an extended pulse with the value stored in ACCU 1, or set a counter (formal operand) with the next count value indicated.
	C	Y	N	Y	46		
SSU=	Formal operand T0 to 127	Y	N	Y	200		Start a stored on-delay timer (formal operand) with the value stored in ACCU 1, or increment a counter (formal operand).
	C	Y	N	Y	36		
SFD=	Formal operand T0 to 127	Y	N	Y	200		Start an off-delay timer (formal operand) with the value stored in ACCU 1, or decrement a counter (formal operand).
	C	Y	N	Y	34		
Load and Transfer Operations							
L=	Formal operand I/Q	N	N	N	52		Load the value of the formal operand into ACCU 1. Parameter type: BY, W
	F				21		
	T0 to 63 T64 to 127				200 37		
	C				27		
	DW, DL, DR				34		
L	RS	N	N	N	6		Load a word from the system data area into ACCU 1.
LD=	Formal operand T0 to 63 T64 to 127	N	N	N	200 60		Load the value of the formal operand in BCD code into ACCU 1.
	C				46		
LW=	Formal operand	N	N	N	18.4		Load a formal operand bit pattern into ACCU 1. Data type: D Parameter type: KC, KF, KH, KM, KS, KT, KY

for organization blocks (OB) for function blocks (FB)
for program blocks (PB) for sequence blocks (SB)

Operation (STL)	Permissible Operands	1 RLO depend. 2 RLO affected 3 RLO reloaded			Typical Execution Time in µs		Function
		1	2	3	On-board	Ext. I/O	
Load and Transfer Operations (cont.)							
T=	Formal operand	N	N	N			Transfer contents of ACCU 1 to the formal operand (Parameter type: BY, W)
	I/Q				52		
	F				20		
	DR, DL, DW				32		
Conversion Operations							
CFW		N	N	N	0.8		Form the one's complement of ACCU 1.
CSW		N	N	N	12.4		Form the two's complement of ACCU 1. CC 1 / CC 0 and OV are affected.
Shift Operations							
SLW	Parameter n=0 to 15	N	N	N	5.6+n · 3.6		Shift the contents of ACCU 1 to the left by the value specified in the parameter. Unassigned positions are padded with zeros. CC 1 / CC 0 are affected.
SRW	Parameter n=0 to 15	N	N	N	5.6+n · 3.6		Shift the contents of ACCU 1 to the right by the value specified in the parameter. Unassigned positions are padded with zeros. CC 1 / CC 0 are affected.
Jump Operations							
JU=	Symb.addr. max. 4 char.	N	N	N	13		Jump unconditionally to the symbolic address.
JC=	Symb. addr. max. 4 char.	Y	Y[1)]	Y	15		Jump conditionally to the symbolic address. (If the RLO is ”0”, it is set to ”1”.)
JZ=	Symb. addr. max. 4 char.	N	N	N	17		Jump if the result is zero. The jump is made only if CC 1=0 and CC 0=0. The RLO is not changed.
JN=	Symb. addr. max. 4 char.	N	N	N	19		Jump if the result is not zero. The jump is made only if CC 1 CC 0. The RLO is not changed.
JP=	Symb. addr. max. 4 char.	N	N	N	17		Jump if the result is greater than 0. The jump is made only if CC 1=1 and CC 0=0. The RLO is not changed
JM=	Symb. addr. max. 4 char.	N	N	N	17		Jump if the result is less than 0. The jump is made only if CC 1=0 and CC 0=1. The RLO is not changed.
JO=	Symb. addr. max. 4 char.	N	N	N	16		Jump on overflow. The jump is made only if the OVERFLOW bit is set. The RLO is not changed.

1) RLO is set to "1".

for organization blocks (OB) for function blocks (FB)
for program blocks (PB) for sequence blocks (SB)

Operation (STL)	Permissible Operands	1 RLO depend. 2 RLO affected 3 RLO reloaded			Typical Execution Time in µs		Function
		1	2	3	On-board	Ext. I/O	
Other Operations							
IA		N	N	N	490		Disable interrupt. Input / output interrupt or timer OB processing is disabled (permiss. for max. 5 ms).
RA		N	N	N	490		Enable interrupt. This operation cancels the effect of IA.
D		N	N	N	1.6		Decrement the low byte (bits 0 to 7) of ACCU 1 by the value n (n=0 to 255).
I		N	N	N	1.2		Increment the low byte (bits 0 to 7) of ACCU 1 by the value n (n=0 to 255).
DO=	Formal operand	N	N	Y	65		Process a block. (Only C DB, JU OB, JU PB, JU FB, JU SB can be substituted.) Actual operands: C DB, JU OB, JU PB, JU FB, JU SB
DO	DW*	N	N	N	15		Process data word. The next operation is combined with the parameter specified in the data word (OR operation) and then carried out.
DO	FW*	N	N	N	20		Process flag word. The next operation is combined with the parameter specified in the flag word (OR operation) and then carried out.

* Permissible operations:
L FY, T FY, L FW, T FW, L IB, T IB, L QB,T QB,
L IW, T IW, L QW, T QW, L DL, T DL, L DR, T DR,
L DW, T DW, JU OB/SB/FB/PB, JC OB/SB/FB/PB/
C DB
A F, S F, R F, = F, SS T,
SE T, R T, A T, AN T,
SLW, SRW

C 1.3 System Operations

Operation (STL)	Permissible Operands	1 RLO depend. 2 RLO affected 3 RLO reloaded			Typical Execution Time in µs		Function
		1	2	3	On-board	Ext. I/O	
Set Operations							
SU	RS	N	N	Y	2.8		Set bit in system data area unconditionally.
RU	RS	N	N	Y	2.8		Reset bit in system data area unconditionally.
Load and Transfer Operations							
LIR		N	N	N	182		Load the contents of a memory word (addressed by ACCU 1) indirectly into the register (0: ACCU 1; 2: ACCU 2).
TIR		N	N	N	50 to 80		Transfer the register contents (0: ACCU 1; 2: ACCU 2) indirectly into the memory word (addressed by ACCU 1).
TNB	Parameter n=0 to 255	N	N	N	200 + n · 6		Transfer a block byte by byte (number of bytes 0 to 255).
T	RS	N	N	N	5.6		Transfer a word to the system data area.
Block Call Operations and Return Operations							
JU	OB	N	N	Y	56		Call an organization block unconditionally.
JC	OB	Y	Y[1)]	Y	56		Call an organization block conditionally.
Arithmetic Operations							
ADD	BN	N	N	N	6		Add byte constant (fixed point) to ACCU 1.
ADD	KF	N	N	N	4.8		Add fixed-point constant (word) to ACCU 1.
Other Operations							
STS		N	N	N	1.2		Stop operation. Program processing is interrupted immediately after this operation.
TAK		N	N	N	4.4		Swap the contents of ACCU 1 and ACCU 2.

C.1.4 Evaluation of CC 1 and CC 0

CC 1	CC 0	Arithmetic Operations	Digital Logic Operations	Comparison Operations	Shift Operations	Conversion Operations
0	0	Result=0	Result=0	ACCU 2=ACCU 1	shifted bit=0	-
0	1	Result<0	-	ACCU 2<ACCU 1	-	Result<0
1	0	Result>0	Result 0	ACCU 2>ACCU 1	shifted bit=1	Result>0

C.2 Machine Code Listing

Machine Code								Operation	Operand
B0		B1		B2		B3			
L	R	L	R	L	R	L	R		
0	0	0	0					NOP 0	
0	1	0	0					CFW	
0	2	0_d	0_d					L	T
0	3	0_l	0_l					TNB	
0	4	0_d	0_d					FR	T
0	5	0	0					BEC	
0	6	0_c	0_c					FR=	
0	7	0_c	0_c					A=	
0	8	0	0					IA	
0	8	8	0					RA	
0	9	0	0					CSW	
0	A	0_a	0_a					L	FY
0	B	0_a	0_a					T	FY
0	C	0_d	0_d					LD	T
0	D	0_i	0_i					JO=	
0	E	0_c	0_c					LC=	
0	F	0_c	0_c					0	
1	0	8	2					BLD	130
1	0	8	3					BLD	131
1	0	8	4					BLD	132
1	0	8	5					BLD	133
1	0	F	F					BLD	255
1	1	0_n	0_n					I	
1	2	0_a	0_a					L	FW
1	3	0_a	0_a					T	FW
1	4	0_d	0_d					SF	T
1	5	0_i	0_i					JP=	
1	6	0_c	0_c					SFD=	
1	7	0_c	0_c					S=	
1	9	0_n	0_n					D	
1	C	0_d	0_d					SE	T
1	D	0_f	0_f					JC	FB

Machine Code								Operation	Operand
B0		B1		B2		B3			
L	R	L	R	L	R	L	R		
1	E	0_c	0_c					SEC=	
1	F	0_c	0_c					==	
2	0	0_f	0_f					C	DB
2	1	2	0					>F	
2	1	4	0					<F	
2	1	6	0					><F	
2	1	8	0					!=F	
2	1	A	0					>=F	
2	1	C	0					<=F	
2	2	0_g	0_g					L	DL
2	3	0_g	0_g					T	DL
2	4	0_d	0_d					SD	T
2	5	0_i	0_i					JM=	
2	6	0_c	0_c					SD=	
2	7	0_c	0_c					AN=	
2	8	0_e	0_e					L	KB
2	A	0_g	0_g					L	DR
2	B	0_g	0_g					T	DR
2	C	0_d	0_d					SS	T
2	D	0_i	0_i					JU=	
2	E	0_c	0_c					SSU=	
2	F	0_c	0_c					ON=	
3	0	0	1	0_e	0_e	0_e	0_e	L	KC
3	0	0	2	0_e	0_e	0_e	0_e	L	KT
3	0	0	4	0_e	0_e	0_e	0_e	L	KF
3	0	1	0	0_e	0_e	0_e	0_e	L	KS
3	0	2	0	0_e	0_e	0_e	0_e	L	KY
3	0	4	0	0_e	0_e	0_e	0_e	L	KH
3	0	8	0	0_e	0_e	0_e	0_e	L	KM
3	2	0_g	0_g					L	DW
3	3	0_g	0_g					T	DW
3	4	0_d	0_d					SP	T

Machine Code								Operation	Operand
B0		B1		B2		B3			
L	R	L	R	L	R	L	R		
3	5	0_i	0_i					JN=	
3	6	0_c	0_c					SP=	
3	7	0_c	0_c					RB=	
3	C	0_d	0_d					R	T
3	D	0_f	0_f					JU	FB
3	E	0_c	0_c					RD=	
3	F	0_c	0_c					LW=	
4	0	0	0_k					LIR	
4	1	0	0					AW	
4	2	0_o	0_o					L	C
4	4	0_o	0_o					FR	C
4	5	0_i	0_i					JZ=	
4	6	0_c	0_c					L=	
4	8	0	0_k					TIR	
4	9	0	0					OW	
4	A	0_a	0_a					L	IB
4	A	8_a	0_a					L	QB
4	B	0_a	0_a					T	IB
4	B	8_a	0_a					T	QB
4	C	0_o	0_o					LD	C
4	D	0_f	0_f					JC	OB
4	E	0_g	0_g					DO	FW
5	0	0_e	0_e					ADD	BN
5	1	0	0					XOW	
5	2	0_a	0_a					L	IW
5	2	8_a	0_a					L	QW
5	3	0_a	0_a					T	IW
5	3	8_a	0_a					T	QW
5	4	0_o	0_o					CD	C
5	5	0_f	0_f	0_e	0_e	0_e	0_e	JC	PB
5	8	0	0					ADD	KF
5	9	0	0					-F	

Machine Code								Operation	Operand
B0		B1		B2		B3			
L	R	L	R	L	R	L	R		
5	C	0_o	0_o					S	C
5	D	0_f	0_f					JC	SB
6	1	0_h	0_h					SLW	
6	2	0_g	0_g					L	RS
6	3	0_g	0_g					T	RS
6	5	0	0					BE	
6	5	0	1					BEU	
6	6	0_c	0_c					T=	
6	9	0_h	0_h					SRW	
6	C	0_o	0_o					CU	C
6	D	0_f	0_f					JU	OB
6	E	0_g	0_g					DO	DW
7	0	0	0					STS	
7	0	0	2					TAK	
7	0	0	3	C	0	0_o	0_o	STP	
7	0	1	5	8	0	0_o	0_o	TB	C
7	0	1	5	4	0	0_o	0_o	TBN	C
7	0	1	5	0	0	0_o	0_o	SU	C
7	0	1	5	C	0	0_d	0_d	RU	C
7	0	2	5	8	0	0_d	0_d	TB	T
7	0	2	5	4	0	0_d	0_d	TBN	T
7	0	2	5	0	0	0_d	0_d	SU	T
7	0	2	5	C	0_b	0_g	0_g	RU	T
7	0	4	6	8	0_b	0_g	0_g	TB	D
7	0	4	6	4	0_b	0_g	0_g	TBN	D
7	0	4	6	0	0_b	0_g	0_g	SU	D
7	0	4	6	C	0_b	0_g	0_g	RU	D
7	0	5	7	8	0_b	0_g	0_g	TB	RS
7	0	5	7	4	0_b	0_g	0_g	TBN	RS
7	0	5	7	0	0_b	0_g	0_g	SU	RS
7	0	5	7					RU	RS

Machine Code								Operation	Operand
B0		B1		B2		B3			
L	R	L	R	L	R	L	R		
7	2	0_a	0_a					L	PB/PY*
7	3	0_a	0_a					T	PB/PY*
7	5	0_f	0_f					JU	PB
7	6	0_c	0_c					DO=	
7	8	0	5	0	0	0_f	0_f	G	DB
7	9	0	0					+F	
7	A	0_a	0_a					L	PW
7	B	0_a	0_a					T	PW
7	C	0_o	0_o					R	C
7	D	0_f	0_f					JU	SB
8	0_b	0_a	0_a					A	F
8	8_b	0_a	0_a					O	F
9	0_b	0_a	0_a					S	F
9	8_b	0_a	0_a					=	F
A	0_b	0_a	0_a					AN	F
A	8_b	0_a	0_a					ON	F
B	0_b	0_a	0_a					R	F
B	8	0_o	0_o					A	C
B	9	0_o	0_o					O	C
B	A	0	0					A(	
B	B	0	0					O(	
B	C	0_o	0_o					AN	C
B	D	0_o	0_o					ON	C

Machine Code								Operation	Operand
B0		B1		B2		B3			
L	R	L	R	L	R	L	R		
B	F	0	0					)	
C	0_b	0_a	0_a					A	I
C	0_b	8_a	0_a					A	Q
C	8_b	0_a	0_a					O	I
C	8_b	8_a	0_a					O	Q
D	0_b	0_a	0_a					S	I
D	0_b	8_a	0_a					S	Q
D	8_b	0_a	0_a					=	I
D	8_b	8_a	0_a					=	Q
E	0_b	0_a	0_a					AN	I
E	0_b	8_a	0_a					AN	Q
E	8_b	0_a	0_a					ON	I
E	8_b	8_a	0_a					ON	Q
F	0_b	0_a	0_a					R	I
F	0_b	8_a	0_a					R	Q
F	8	0_d	0_d					A	T
F	9	0_d	0_d					O	T
F	A	0_i	0_i					JC=	
F	B	0	0					O	
F	C	0_d	0_d					AN	T
F	D	0_d	0_d					ON	T
F	F	F	F					NOP 1	

* depending on the type of programmer used

Explanation of the Indices

a	**+ byte address**	**h**	**+ number of shifts**
b	**+ bit address**	**i**	**+ relative jump address**
c	**+ parameter address**	**k**	**+ register address**
d	**+ timer number**	**l**	**+ block length in bytes**
e	**+ constant**	**m**	**+ jump displacement (16 bits)**
f	**+ block number**	**n**	**+ value**
g	**+ word address**	**o**	**+ counter number**

C.3 List of Abbreviations

Abbreviation	Explanation	Permissible Operand Value Range for S5-95F
ACCU 1	Accumulator 1 (When accumulator 1 is loaded, any existing contents are shifted into accumulator 2.)	
ACCU 2	Accumulator 2	
BN	Byte constant (fixed-point number)	(- 127 to +127)
C	Counter - retentive - non-retentive - for the ”Bit Test” and ”Set” supplementary operations	 (0 to 7) (8 to 127) (0 to 127) (0.0 to 127.15)
CAN	DB1 parameter: counter A counts with negative and positive edge	
CAP	DB1 parameter: counter A counts with positive edge	
CBN	DB1 parameter: counter B counts with negative and positive edge	
CBP	DB1 parameter: counter B counts with positive edge	
CC 0 / CC 1	Condition code 0 / Condition code 1	
CCN	DB1 parameter: cascaded counter counts with negative edge	
CCP	DB1 parameter: cascaded counter counts with positive edge	
CF	DB1 parameter: input correction factor (integral real-time clock)	
CLK	DB1 parameter: clock data location	
CPU	Central processing unit of programmable controller	
CSF	STEP 5 control system flowchart method of representation	
D	Data (1 bit)	(0.0 to 255.15)
DB	Data block	(2 to 255)
DL	Data word (left-hand byte)	(0 to 255)
DR	Data word (right-hand byte)	(0 to 255)
DW	Data word	(0 to 255)
EF	DB1 parameter: SINEC L1, position of receive mailbox	
F	Flag - retentive - non-retentive	(0.0 to 63.7) (64.0 to 255.7)
FB	Function block	(0 to 255)
FB/FY	Flag byte - retentive - non-retentive	(0 to 63) (64 to 255)
Formal operand	Expression with a maximum of 4 characters. The first character must be a letter of the alphabet.	

Abbreviation	Explanation	Permissible Operand Value Range for S5-95F
FW	Flag word - retentive - non-retentive	(0 to 62) (64 to 254)
I	Input	(0.0 to 127.7)
IB	Input byte	(0 to 127)
IW	Input word	(0 to 126)
KB	Constant (1 byte)	(0 to 255)
KBE	DB1 parameter: SINEC L1, position of the ”Receive” coordination byte	
KBS	DB1 parameter: SINEC L1, position of the ”Send” coordination byte	
KC	Constant (count)	(0 to 999)
KF	Constant (fixed-point number)	(- 32768 to +32767)
KH	Constant (hexadecimal code)	(0 to FFFF)
KM	Constant (2-byte bit pattern)	(arbitrary bit pattern: 16 bits)
KS	Constant (2 characters)	(any two alphanumeric characters)
KT	Constant (time)	(0.0 to 999.3)
KY	Constant (2 bytes)	(0 to 255 per byte)
LAD	STEP 5 ladder diagram method of representation	
NT	DB1 parameter: number of timers being processed	
OB	Organization block for special applications: 1, 2, 3, 13, 21, 22, 31, 34, 37, 251	(0 to 255)
OB13	DB1 parameter: interval (ms) within which OB13 is called and processed	
OBC	DB1 block ID for onboard counters	
OHE	DB1 parameter: enable operating hours counter	
OHS	DB1 parameter: set operating hours counter	
OP	Operator panel	
OV	Overflow. This condition code bit is set if, e.g., a numerical range is exceeded during arithmetic operations.	
PB	Program block (with block call and return operations)	(0 to 255)
PB or PY (depending on type of programmer used)	Peripheral byte	(0 to 127)
PG	Programmer	
PGN	DB1 parameter: SINEC L1, programmer bus number	
PII	Process image input table	

Abbreviation	**Explanation**	**Permissible Operand Value Range for S5-95F**
PIQ	Process image output table	
PW	Peripheral word	(0 to 126)
Q	Output	(0.0 to 127.7)
QB	Output Byte	(0 to 127)
QW	Output word	(0 to 126)
RLO	Result of logic operation	
RLO affected? Y/N	The RLO is affected/not affected by the operation.	
RLO dependent? Y Y /Y N	The statement is executed only if the RLO is ”1”. The statement is executed only on positive/negative edge change of the RLO. The statement is always executed.	
RLO reloaded? Y/N	When the next binary operation takes place, the RLO is reloaded/not reloaded (e.g. A I 0.0).	
RS	System data area - for load operations (supplementary operations) and transfer operations (system operations) - for bit test and set operations (system operations)	(0 to 255) (0.0 to 255.15)
SAC	STEP address counter	
SAV	DB1 parameter: Save clock setting after last STOP-to-RUN transition or POWER OFF	
SB	Sequence block	(0 to 255)
SDP	DB1 block ID for system data parameters	
SF	DB1 parameter: SINEC L1, position of send mailbox	
SL1	DB1 block ID for SINEC L1	
SLN	DB1 parameter: SINEC L1, slave number	
STL	STEP 5 statement list method of representation	
STP	DB1 parameter: update the clock while in the STOP state.	
STW	DB1 parameter: status word location (integral real-time clock)	
T	Timer - for the ”Bit Test” and ”Set” supplementary operations	(0 to 127) (0.0 to 127.15)
TD	Text display	
TFB	DB1 block ID for timer function block	
TIS	DB1 parameter: set prompt time	

D Guidelines for Handling Electrostatic Sensitive Devices (ESD)

Figures

D-1 ESD Measures . . . . . D - 4

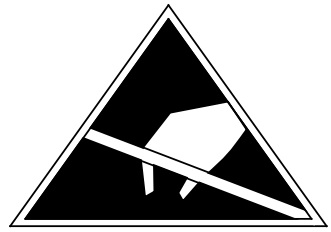
D Guidelines for Handling Electrostatic Sensitive Devices (ESD)

What is ESD?

All electronic modules are equipped with large-scale integrated ICs or components. Due to their design, these electronic elements are very sensitive to overvoltages and thus to any electrostatic discharge.

These **E**lectrostatic **S**enstive **D**evices are commonly referred to by the abbreviation **ESD.**

Electrostatic sensitive devices are labelled with the following symbol:



Caution

Electrostatic sensitive devices are subject to voltages that are far below the voltage values that can still be perceived by human beings. These voltages are present if you touch a component or module without previously being electrostatically discharged. In most cases, the damage caused by an overvoltage is not immediately noticeable and results in total damage only after a prolonged period of operation.

Electrostatic charging of objects and persons

Every object with no conductive connection to the electrical potential of its surroundings can be charged electrostatically. In this way, voltages up to 15000 V can build up whereas minor charges, i.e. up to 100 V, are not relevant.

Examples:

• Plastic covers	up to 5000 V
• Plastic cups	up to 5000 V
• Plastic-bound books and notebooks	up to 8000 V
• Desoldering device with plastic parts	up to 8000 V
• Walking on plastic flooring	up to 12000 V
• Sitting on a padded chair	up to 15000 V
• Walking on a carpet (synthetic)	up to 15000 V

Limits for perceiving electrostatic discharges

An electrostatic discharge is

- perceptible from 3500 V
- audible from 4500 V
- visible from 5000 V

A fraction of these voltages is capable of destroying or damaging electronic devices.

Carefully note and apply the protective measures described below to protect and prolong the life of your modules and components.

General protective measures against electrostatic discharge damage

- Keep plastics away from sensitive devices. Most plastic materials have a tendency to build up electrostatic charges easily.

- Make sure that the personnel, working surfaces and packaging are sufficiently grounded when handling electrostatic sensitive devices.

- If possible, avoid any contact with electrostatic sensitive devices. Hold modules without touching the pins of components or printed conductors. In this way, the discharged energy cannot affect the sensitive devices.

Additional precautions for modules without housings

Note the following measures that have to be taken for modules that are not protected against accidental contact:

- Touch electrostatical sensitive devices only
  - if you wear a wristband complying with ESD specifications or
  - if you use special ESD footwear or ground straps when walking on an ESD floor.

- Persons working on electronic devices should first discharge their bodies by touching grounded metallic parts (e.g. bare metal parts of switchgear cabinets, water pipes, etc.).

- Protect the modules against contact with chargeable and highly insulating materials, such as plastic foils, insulating table tops or clothes made of plastic fibres.

- Place electrostatic sensitive devices only on conductive surfaces:
  - Tables with ESD surface
  - Conductive ESD foam plastic (ESD foam plastic is mostly coloured black)
  - ESD bags

- Avoid direct contact of electrostatic sensitive devices with visual display units, monitors or TV sets (minimum distance to screen > 10 cm).

The following Figure once again illustrates the precautions for handling electrostatically sensitive devices.

a Conductive flooring material
b Table with conductive, grounded surface
c ESD footwear
d ESD smock
e Grounded ESD wristband
f Ground connection of switchgear cabinet
g Grounded chair

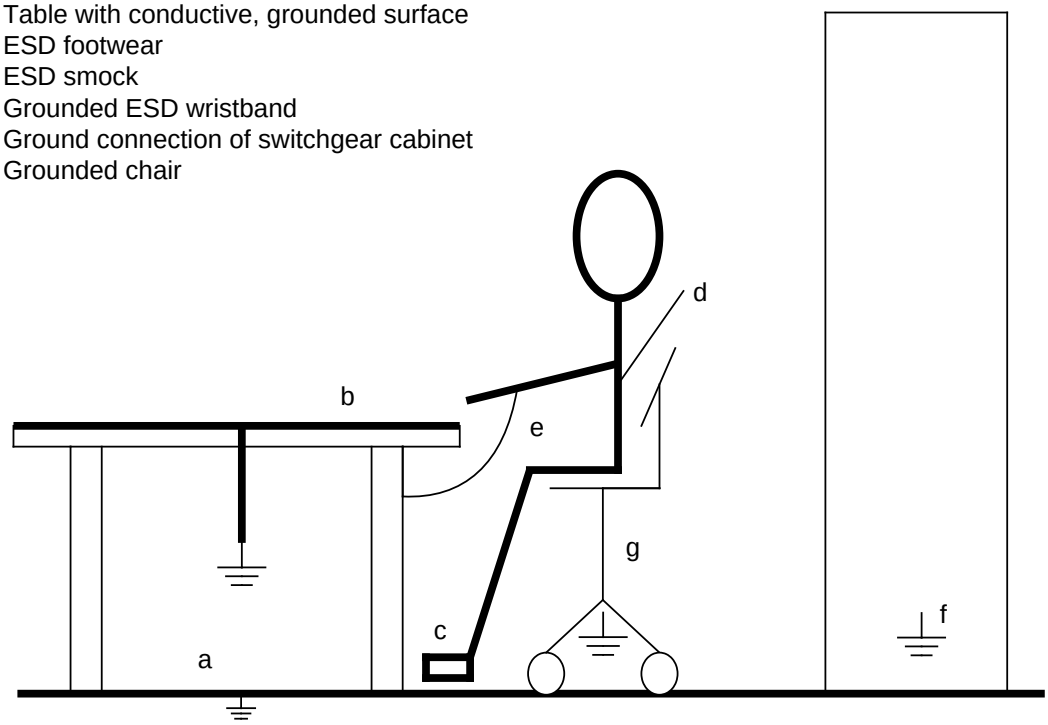


Figure D-1. ESD Measures

Taking measurements and working on ESD modules

Measurements may be taken on electrostatic sensitive devices only if

- the measuring device is grounded (e.g. via protective conductor) or
- the tip of the isolated measuring tool has previously been discharged (e.g. by briefly touching grounded metal parts).

E Prototype Test Certification

E Prototype Test Certification

The S5-95F has been tested by the Technical Inspectorate of Bavaria (TÜV-Bayern) and the Institute for Industrial Safety of the Statutory Industrial Accident Insurance Institution (BIA).

Copies of the report and the certificates issued in connection with the prototype tests can be obtained from us on request. Please direct inquiries to:

Siemens AG
AUT 125
attn. Mrs. Bleicher

P.O. Box 1963

D-92209 Amberg
Federal Republic of Germany

Telefax: ++49- 9621-803146

Index

Index

A

AC power cable 3-4
Acceptance 18-1
- test 18-5
Access
- to the process image **6-10f**
Accident prevention
- rules 3-1
Accumulator **6-10f**, **6-14f**
Actual operand **7-14ff**
Actual parameter 7-11
Actuator
- requirements 18-20
- tested 18-20
Actuator triggering **4-19ff**
Address **18-35f**
- area 6-5
- area restrictions 8-65
- assignment **6-1ff**, 6-5
- assignments in RAM 6-15
Addressing **18-35f**
Analog input module **11-7ff**
Analog module
- addressing 6-5, 6-7
Analog output module **11-10ff**
Analog value
- conversion block 9-55, **11-11ff**
- non-failsafe processing **11-1ff**
Analog value processing 9-55
- circuit versions **11-39f**
- failsafe **11-18ff**
Applications 17-1
- regulations **1-6f**
- typical 1-6
Arithmetic operation **8-30f**
Arithmetic unit 2-5, 2-7

B

Basic insulation 3-5
Basic operation 7-2, **8-1ff**
Basic system 2-2, 4-1, 5-1
- hardware overview 1-8
Basic unit **4-2ff**, **5-1f**, 5-7, 5-10
- with external I/Os 2-2
- without external I/Os 2-1
Battery 18-18
- compartment 2-1, 2-11
- disposal 2-19
- failure 2-11, 7-9, **9-2**
- failure display 2-1
- replacement 2-19, 18-18
Binary scaler 8-70
Bit
- address 6-2
Blanking time 4-21
Block
- address list 6-17
- call 7-11, 7-17, 8-34
- call operations 9-1
- header **7-12f**
- name 7-12
- nesting depth 7-27
- parameter **7-12f**, **7-15**
- stack 7-6, 14-12
- types **7-7f**, 7-11
Block call operation **8-32f**
Boolean logic operation **8-2f**, 9-1
BSTACK 14-12
BT777 Bus terminal BT777
Burner
- control 1-4
Bus 18-3
- monitoring 2-8, 13-23
- number 13-5, 13-13
- unit 2-8, **5-1ff**
Bus terminal BT777 **13-1f**
Bus unit 2-2

C

Cabinet
- door 3-5, 3-14
- lighting 3-12
- mounting 5-7
- panel 3-4
- ventilation 5-6
Cable
- clamps 3-11
- duct 5-7
- route 3-7
- shielding 3-14
- tray 3-8

Cables
- routing of **3-7f**
Call
- disabling 7-27
Call up interval 6-13
- OB13 **18-26f**
Channel
- number 6-1, 6-4
Checklist
- for EMC installation **3-14**
Circuit diagrams 18-7
Circulating current 3-10
Clock **10-1ff**
- data 2-18, 7-18, **10-2ff**
- data area 6-18
- parameters 10-2
- time correction factor 10-2
Clock-pulse generator 8-72
Code converter 9-53
Coding element 5-3
Cold restart routine 7-19
COM 95F **18-1f**, **18-4f**
Communication 18-3
- byte 2-4, **6-8f**
- partner 18-3
Comparison
- operation **8-29f**
Compiler
- error 14-10
Connecting cable 712 **5-5f**
Connection method 5-9
Connector pin assignment 4-18, 4-30
- counter 12-2
- onboard I/Os **4-31ff**
- onboard interrupt DI 12-2
Contact washer 3-4, 3-14
Contactor 3-11, 18-20
Control
- deviation 9-9
Control circuit 5-27, **5-27ff**
Control functions
- programmer **14-13f**
Control System Flowchart (CSF) 1-9, 7-7, **7-1ff**
Controlled system 9-10
Controls 2-1, 2-11
Conversion operation **8-49ff**
Coordination byte 13-12, 13-16
- for SINEC L1 **13-4ff**, **13-9f**
Core end sleeve 5-9
Counter 18-19
- for counting 4-25
- for rotational speed monitoring **4-27ff**
- parameterization 4-25
- for frequency monitoring **4-27ff**
- input **4-23ff**
- operation 8-25
- scanning 8-26
Counter status
- scanning and resetting 4-29
Counting frequency 4-24
Counting range 8-25
Coupling element **4-21f**
Coupling path 3-3
Coupling relay **4-21f**, 18-20
- connection **5-18f**
CP 521 SI 18-3
CPU 4-1
Crimp-snap-in
- connection method 2-2
CSF Control System Flowchart
Current
- sensor 11-4
Cycle
- monitoring 2-17, 7-19, 14-10
- trigger 7-21, 14-1
Cycle time
- exceeding 9-56, 14-10
- loading 7-22
- monitoring 7-21
- statistics 7-23, 15-1, 15-21
- triggering 7-9, **9-2**

D

Damage
- extent 1-3
Danger
- prevention 1-3
Data
- block 18-19
- cycle **6-10f**, 7-23, 7-28
- cycle time 2-10
- interchange 18-3
- interchange, failsafe 13-11
- interchange, non-failsafe 13-3
- path **13-22ff**
- transfer 18-3

Data block 7-1, 7-3, **7-16f**, 7-27
- DB1 **9-63f**
- generating and deleting **8-34f**
- retentive characteristics **2-7**
Data cycle 2-9
Data path
- failsafe **18-46**
DB Data block
Depassivation
- block 15-19
- block FB255 **9-61f**
- SINEC L1 13-22
Diagnostic byte 2-4, **6-8f**, 6-16, 12-3, **12-5f**, 12-11
- for battery and load voltage 15-22
Digital input
- availability 9-15
- non equivalence check 9-34
Digital module
- addressing **6-4f**
- connection **5-29f**
Digital modules
- connection 5-23
Digital outputs
- testing with FB252 4-22
Direct access 2-17
- to onboard I/Os **6-12f**
- to external I/Os
Discrepancy
- analysis 4-8, 18-15
- time 4-8, 4-10, 18-15
Discrepancy analysis
- for analog values **11-19f**
Display generation
- operation 8-38
Displays
- optical 2-12
Divider 9-54
”Do” operation **8-53f**

E

Earth fault monitoring 5-32
Edge evaluation 8-70
EMC **3-1ff**
- checklist **3-14**
Enable operation 8-40
EPROM **2-14ff**
- programming 14-15
EPROM Memory submodule
Equipotential bonding 3-9
Error
- address 14-10, **14-10f**
- block 15-3, **15-8ff**
- entry 15-3, 15-19
- frequency 15-5
- indication 15-2
- indicator **15-1f**
- location 15-4
- message 18-15
- reports 15-11, 18-21
- response 15-1
- response, SINEC L1 13-30
- stack 15-8
Error block
- counter 15-8
Error group
- image 15-5
Error handling
- OB 37 **9-2**
Error messages
- printout 15-20
- to SINEC L1 master 13-31
Error response
- SINEC L1 18-44
Expansion
- tier 5-6
External I/O module
- connection 5-1, 5-5, **5-13ff**
External power supply unit 3-3

F

Failsafe I/O module
- connection **5-12ff**
Failsafety
- objectives 18-2
Fault
- analysis 14-6
- cause 14-5
FB Function block
Fiber optic cable 2-1, 2-2, 2-12, 4-1
- setting the length **4-5f**
Field transfer 8-65
Filter
- for power supply units 3-13
- program 7-18
First scan 8-2
Flag **2-5ff**, 6-16, 18-19
- area **18-31f**
Flip-flop **8-8f**
Fluorescent lamp 3-4, 3-12
FO Fiber optic

FORCE VAR 14-4
Formal operand 7-12
Frame
- loss 13-16
- useful 13-10
Freewheel diode 3-11
Frequency
- range 3-10
- monitoring **4-27f**
Front view
- DI/DO 482 5-25
Function block **7-3ff**, 7-11, **7-14f**, 9-1, 18-4, **18-38f**
- address list 6-17
- integrated **9-14ff**
Functional units 2-5

G

Galvanic isolation
- of external I/Os 5-29
Ground
- connection 5-29
Ground fault monitoring 5-32
Ground loop **3-4ff**
Grounded configuration 5-31
Grounding strip 3-5, 3-14

H

Heat
- dissipation 5-6, 5-8
Height 5-10
High-availability digital inputs 9-15
High-availability S5-95F **16-1ff**
Hydraulic lifting device 17-1

I

I/O areas
- in the process image 6-9
I/O bus
- mode of operation **2-8f**
I/O module
- connection **5-23ff**
I/Os
- access 2-8
- error 4-7
Incandescent bulb 3-4
Inductance 3-4, 3-11, 3-14
Input
- circuits 5-27
- module 4-8, 5-12, 5-23
Input data cycle Data cycle
Inspection
- conditions of 3-1
Insulation monitor 5-32
Interface module 5-1, **5-5ff**
- unused connectors 5-5
Interference source 3-2, 3-10
Interpolation block 9-41
Interrupt
- analysis 14-5
- call programming 12-6
- condition code word 14-7
- data cycle 7-28
- disable 7-27, 18-31
- disable/enable 8-52
- error 7-26, 7-28
- frame 13-3, 13-11
- input **12-1ff**
- list 18-44
- load **7-21f**
- priority 12-1, 12-3, 12-10
- processing 4-23, 7-29, **12-1ff**, 18-27
- programming response **12-13f**
- response time 12-1, **12-15f**, **18-29ff**
- stack 6-18, **14-5ff**
Interrupt data cycle Data cycle
Interrupt DI
- connecting **12-7ff**
ISTACK **14-5ff**

J

Jump
- displacement 8-56
- operation 7-5, 7-21, **8-55f**

K

KBE Coordination byte
KBS Coordination byte

L

LAD Ladder Diagram
Ladder Diagram (LAD) 1-9, 7-1, **7-1ff**, 7-7
Lightning protection
- filter 3-8
- measures 3-8
Limit frequency
- onboard counters 4-23
Line monitoring **4-11f**
Load circuit **5-29f**

Load operations **8-10f**
Load voltage 4-3
- failure 9-57
Logic operation
- result of (RLO) 6-11
Logical program counter 9-55
- test 9-57

M

Mains buffering 4-4
Memory submodule 14-1, **18-18f**
- EPROM **2-5f**, 2-20
Message mode 13-10
Method of representation
- STEP 5 **7-1ff**
Minimum clearance 5-8
- for routing of cables **3-7**
Minimum signal duration
- for interrupt inputs **12-15f**
Monitor 3-7
Monitoring
- time 7-21
Mounting plates 5-6
Multiplier 9-54

N

Nesting
- depth 7-6
”No” operation 8-37
Non-equivalence check 9-34
Non-grounded configuration 5-32
Number
- formats **7-31f**
- representation 7-31

O

OB13
- call-up interval **7-25f**
Old value 15-1
ON/OFF switch 2-1, 2-11
Onboard counter inputs
- connection **4-9f**, 4-24
Onboard digital outputs
- connection **4-18f**
Onboard I/Os **2-2ff**, 4-1, **4-9ff**
- addressing **6-1f**, **6-8f**, **6-12f**
- connector pin assignment 4-23, **4-31ff**
- error indication 15-6
Onboard interrupt **12-1ff**
- I/Os 4-23
OP Operator panel
Operand area 7-3
Operating hours counter 10-1, **10-8f**, **10-18f**
Operating mode 18-17
- display 2-1
- selector switch 2-1
Operating statuses
- following power-up 2-17
Operating system 2-6
- runtime **7-23f**
Operator panel (OP) 2-1, 9-23, 18-43
Organization block 7-4, 7-6, **7-9f**, 7-25
Output
- circuits 3-11, 5-27
- module 5-12
Output data
- cycle 7-28
Output data cycle Data cycle
Overall reset **2-18**, 18-26
Overvoltage 3-8

P

Parameter
- list **7-15f**
Parameter assignment
- operating system **18-1f**, 18-17
Parameter block
- for SINEC L1 13-4, 13-12
Parameterization software COM 95F 1-9
Passivation 4-7, **15-1f**, **18-22ff**
- SINEC L1 13-14
PB Program block
PG Programmer
PID controller **9-3ff**
- mode 9-10
- parameters 9-3
- sampling interval 9-9
PII Process image
PIQ Process image
Planning phase **18-2f**
Plugging in and removing
- modules 4-3
Polling
- list 18-44
- time 18-44
Potential bonding
- of external I/Os 5-29
Power failure 2-19, 7-31
- monitoring 4-30

Power supply
- for S5-95F **4-4f**, 5-27
- module 2-3, 5-2
Power supply unit 3-13
- filter for 24 V DC 3-13
Pre-acceptance inspection 18-4
Press
- control 1-5
Probability of occurence
- risks 1-2
Process
- image **2-5ff**, 6-6, **6-8ff**
- interrupt 12-1
Product
- identification 15-3
PROFIBUS 13-34
Program
- cycle 6-14, 7-21
- processing **18-27ff**
- structure 7-4
- test 7-5
Program block (PB) 7-2, 7-5, **7-7f**, 7-11, 7-14, 7-30, 9-1
- address list 6-17
Programmer (PG) 7-1, **7-15f**, **7-18f**, **7-30f**
- control functions **14-13f**, 18-41
- functions 18-41
- interface for 2-1
Protective conductor 3-4, 3-6, 3-10,
- bar 3-14
Prototype test 1-1
Pulse train 8-72

Q

Quality level **1-2ff**, 1-4
Quasi-safety mode **2-15**
Quenching element 3-11

R

Rack
- mounting **3-6**
Receive
- coordination 13-21
- data 13-21
- mailbox **13-4ff**, 18-44
Redundant structure 1-9
Reference potential
- interrupt input 12-2
Regulations
- applications **1-6f**
Relay 18-20
Repairs 18-26
Response
- S5-95F 18-15, 18-21
Response time
- basic 7-26, 18-3
- for cyclical program processing **7-23f**, **18-27f**
- for interrupt-driven processing 18-29
- for time-controlled program processing **18-28f**
- interrupt processing 12-1, 12-14
- SINEC L1 18-50
Restrictions
- in programming 18-31
Result of logic operation (RLO) 7-4, 8-68
Result of scan 8-2
Retentive characteristics 2-7
Retentivity
- counters 18-19
- data blocks 18-19
- flags 18-19
- timers 18-19
Retriggering 7-21
Risk
- analysis 18-2
- graph 1-2
- parameter 1-2
- potential 1-1
- probability of occurence 1-2
RLO Result of logic operation
Rotational speed monitoring **4-27f**
Routing of cables **3-7f**
Runtime
- errors **14-10f**

S

S5-95F
- high-availability **16-1ff**
Safety
- class 1-1
- mode 2-13, **2-16**, 7-1, 7-16, 7-18, 7-30, 15-19
- regulations 1-1
Safety time 13-14
Safety time
- SINEC L1 **13-11f**, **18-43ff**
Scratchpad 18-32

Screw-type terminal 5-9
Self-tests 2-13
Send
- coordination 13-19
- mailbox **13-4ff**, 13-15, **13-18ff**
Sensor
- circuits 5-27
- connection to DI/DO 482 5-25
- requirements 18-19
Sequence block **7-5f**, 7-11, 7-14, 9-1
Sequence control 7-5, 7-11
Sequencer 9-1
Set operation 8-63
Set/reset operations **8-7f**, 9-1
Shield
- bar 3-6, **3-10ff**
- connection 3-10, 3-11
Shift operations 8-47
Shift register
- length 2-10
Short-circuit
- test **4-11f**, **9-58ff**
Signal group 4-7, 4-18, **18-21ff**
- image 15-5
SINEC L1 2-1, **13-1ff**, 18-3, 18-44
- condition code and control byte 13-22
- error image 15-7
- forwarding error messages **15-20f**
- response times 18-50
- safety times 13-25
Slot 18-36
Slot
- number 2-8, **6-1ff**, **6-3f**
Spark gap 3-3
Standard mounting rail 2-2, **4-2ff**, **5-1f**
Standard value
- formation 15-19, 18-25
Start operation 8-40
Startup
- behavior 2-13, 2-17
- block **7-18ff**
- procedures 7-9
Statement list (STL) **7-1ff**
STATUS **14-1f**
STATUS VAR 14-3
STEP 5
- method of representation **7-1ff**
Step response 9-9
Structural elements 3-6
Submodule
- order number 2-20
Subprocess **18-2f**, 18-23
Subroutine
- technique 7-5
Substitution operation 8-57
Subunit **2-14ff**
Subunit identifier
- setting **4-6f**
- switch 2-12
Supply voltage 18-26
- monitoring 4-30
Surge impulse 3-8
System
- data 6-18
- initialization 2-18
System
- operation 8-64, 9-1
- parameter assignment **9-63f**
- response 15-4, 18-21, 18-45
System data 2-7
- area 6-15, 6-18
- loading 8-39
System event DB 15-3
System events 15-11
System response 4-7

T

Terminal block 6-1
Test
- component 9-56, 9-57
- function 14-1
- interval 9-55
- mode 2-14, 18-17
- operations 8-41
Test routine 9-55
- calling 9-55
Text display 9-23, 18-43
Third-party system 3-7
Time
- control 7-9
- stamp 10-14

Time interrupt
- processing **7-25ff**
Timer
- area **8-13f**
- off-delay 8-15, 8-24
- on-delay **8-22f**
- operation 8-15
- state 8-15
Touch voltage 3-5
Transducer
- four-wire 11-5
- two-wire 11-5
Transfer
- error 14-10
- operation **8-10f**, 9-1
Transformer 3-3
Two-way radio 3-3

U

User
- reaction 4-7
User program
- securing against errors 14-15
User valid byte (UVB) **13-11ff,** 13-19, 13-21
UVB User valid byte

V

Voltage
- dip 4-3
- range setting 3-1
- rated 4-4
- sensor 11-3

W

Wall mounting Rack mounting
Wire clamp 5-9
Wiring
- high-availability S5-95F 16-2

Siemens AG
AUT E 148
Postfach 1963

D-92209 Amberg
Federal Republic of Germany

From:

Your Name: _____

Your Title: _____

Company Name: _____

Street: _____

City, Zip Code: _____

Country: _____

Phone: _____

Please check any industry that applies to you:

- ☐ Automotive
- ☐ Chemical
- ☐ Electrical Machinery
- ☐ Food
- ☐ Instrument and Control
- ☐ Nonelectrical Machinery
- ☐ Petrochemical
- ☐ Pharmaceutical
- ☐ Plastic
- ☐ Pulp and Paper
- ☐ Textiles
- ☐ Transportation
- ☐ Other _____

Remarks Form

Your comments and recommendations will help us to improve the quality and usefulness of our publications. Please take the first available opportunity to fill out this questionnaire and return it to Siemens.

Title of Your Manual: _____

Order No. of Your Manual: _____

Edition: _____

Please give each of the following questions your own personal mark within the range from 1 (very good) to 5 (poor).

1. Do the contents meet your requirements? ☐
2. Is the information you need easy to find? ☐
3. Is the text easy to understand? ☐
4. Does the level of technical detail meet your requirements? ☐
5. Please rate the quality of the graphics/tables: ☐

Additional comments:

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____