

SIEMENS

SIMATIC

Industrial software S7 F/FH Systems - Configuring and Programming




Programming and Operating Manual

| | |
|---|----|
| Security information | 1 |
| Preface | 2 |
| Product Overview | 3 |
| Installing | 4 |
| Configuration | 5 |
| Access Protection | 6 |
| Programming | 7 |
| F-I/O access | 8 |
| Programming communication | 9 |
| Operator inputs with the "Secure Write Command++" function | 10 |
| Safety Data Write function: Changing F-parameters from the OS | 11 |
| Compiling & commissioning an S7 program | 12 |
| System Acceptance Test | 13 |
| Operation and Maintenance | 14 |
| F-libraries | A |
| Checklist | B |
| Requirements for virtual environments and remote access | C |

Legal information

Warning notice system

This manual contains notices you have to observe in order to ensure your personal safety, as well as to prevent damage to property. The notices referring to your personal safety are highlighted in the manual by a safety alert symbol, notices referring only to property damage have no safety alert symbol. These notices shown below are graded according to the degree of danger.

| |
|--|
|  DANGER |
| indicates that death or severe personal injury will result if proper precautions are not taken. |
|  WARNING |
| indicates that death or severe personal injury may result if proper precautions are not taken. |
|  CAUTION |
| indicates that minor personal injury can result if proper precautions are not taken. |
| NOTICE |
| indicates that property damage can result if proper precautions are not taken. |


If more than one degree of danger is present, the warning notice representing the highest degree of danger will be used. A notice warning of injury to persons with a safety alert symbol may also include a warning relating to property damage.

Qualified Personnel

The product/system described in this documentation may be operated only by **personnel qualified** for the specific task in accordance with the relevant documentation, in particular its warning notices and safety instructions. Qualified personnel are those who, based on their training and experience, are capable of identifying risks and avoiding potential hazards when working with these products/systems.

Proper use of Siemens products

Note the following:

| |
|--|
|  WARNING |
| Siemens products may only be used for the applications described in the catalog and in the relevant technical documentation. If products and components from other manufacturers are used, these must be recommended or approved by Siemens. Proper transport, storage, installation, assembly, commissioning, operation and maintenance are required to ensure that the products operate safely and without any problems. The permissible ambient conditions must be complied with. The information in the relevant documentation must be observed. |

Trademarks

All names identified by ® are registered trademarks of Siemens AG. The remaining trademarks in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owner.

Disclaimer of Liability

We have reviewed the contents of this publication to ensure consistency with the hardware and software described. Since variance cannot be precluded entirely, we cannot guarantee full consistency. However, the information in this publication is reviewed regularly and any necessary corrections are included in subsequent editions.

Table of contents

| | | |
|----------|---|-----------|
| 1 | Security information | 11 |
| 2 | Preface | 13 |
| 2.1 | Warnings index | 19 |
| 3 | Product Overview..... | 23 |
| 3.1 | Overview | 23 |
| 3.2 | Hardware and software components | 26 |
| 4 | Installing | 29 |
| 4.1 | Installing the S7 F Systems optional package | 29 |
| 4.2 | Uninstalling the S7 F Systems optional package | 31 |
| 4.3 | Migration to S7 F Systems V6.2 | 32 |
| 4.3.1 | User scenario 1 | 34 |
| 4.3.2 | User scenario 2 | 35 |
| 4.3.3 | User scenario 3 | 39 |
| 4.3.4 | User scenario 4 | 42 |
| 4.3.5 | Updating custom F-block types | 43 |
| 4.3.6 | Updating a multiproject master data library | 43 |
| 5 | Configuration | 45 |
| 5.1 | Configuration overview | 45 |
| 5.2 | Particularities for configuring an F-System | 46 |
| 5.3 | Configuring the F-CPU..... | 47 |
| 5.4 | Configuring the F-I/O | 49 |
| 5.4.1 | Overview of configuring the F-I/O | 49 |
| 5.4.2 | Configuring the fail-safe modules with assignment of F_destination_address via DIP switch | 49 |
| 5.4.3 | Configuring fail-safe modules with assignment of F_destination_address in the Engineering System..... | 52 |
| 5.4.3.1 | Identifying fail-safe modules (address assignment in the ES)..... | 53 |
| 5.4.3.2 | Assigning F-destination address and F-source address (address assignment in the ES) | 55 |
| 5.4.3.3 | Changing the F-destination address or F-source address (address assignment in the ES) | 55 |
| 5.5 | Configuring fail-safe DP standard slaves/IO standard devices | 56 |
| 5.6 | Configuring fail-safe PA field devices | 61 |
| 5.7 | Configuring redundant F-I/O | 62 |
| 5.8 | System modifications during operation..... | 64 |
| 5.8.1 | Configuring F-I/O with CiR | 65 |
| 5.8.2 | Configuration in RUN im H-System (H-CiR)..... | 67 |

| | | |
|----------|--|------------|
| 6 | Access Protection | 69 |
| 6.1 | Overview of access protection | 69 |
| 6.2 | Setting up access rights for the F-CPU | 71 |
| 6.3 | Setting up access permission for the safety program..... | 73 |
| 7 | Programming | 77 |
| 7.1 | Overview of programming | 77 |
| 7.1.1 | Structure of the safety program | 78 |
| 7.2 | Creating the Safety Program | 80 |
| 7.2.1 | Basic procedure for creating the safety program..... | 80 |
| 7.2.2 | Defining the program structure | 81 |
| 7.2.3 | Assigning parameters for the maximum F-cycle monitoring..... | 81 |
| 7.2.4 | Rules for programming | 82 |
| 7.2.5 | Notes for working with CFC | 83 |
| 7.2.6 | Inserting CFC charts | 83 |
| 7.2.7 | Inserting F-Runtime groups | 84 |
| 7.2.8 | F-Shutdown groups..... | 84 |
| 7.3 | Inserting and interconnecting F-Blocks..... | 86 |
| 7.3.1 | Inserting F-Blocks | 86 |
| 7.3.2 | Parameter assignment and interconnection of F-Blocks | 86 |
| 7.3.3 | Determining the runtime sequence | 87 |
| 7.4 | Automatically inserted F-Blocks..... | 89 |
| 7.5 | F-Startup and reprogramming restart/startup protection | 91 |
| 7.6 | F-STOP | 93 |
| 7.7 | Creating F-Block types..... | 95 |
| 7.7.1 | Introduction | 95 |
| 7.7.2 | Rules for F-Block types | 95 |
| 7.7.3 | Creating F-Block types with "Compile Chart as F-Block Type" | 97 |
| 7.7.4 | Modifying F-Block types..... | 99 |
| 7.7.5 | Integrating F-parameters of custom F-block types in the printout of the safety program | 99 |
| 7.8 | Programming data exchange between F-Shutdown groups in an F-CPU..... | 101 |
| 7.9 | Data exchange between safety program and standard user program | 103 |
| 7.9.1 | Programming data exchange from the safety program to the standard user program..... | 103 |
| 7.9.2 | Programming data exchange from the standard user program to the safety program..... | 104 |
| 7.10 | Implementation of user acknowledgment | 106 |
| 8 | F-I/O access | 109 |
| 8.1 | Positioning, interconnecting, and assigning parameters to F-Channel drivers | 110 |
| 8.2 | Generating F-Module drivers | 111 |
| 8.3 | Process data or fail-safe values..... | 112 |
| 8.4 | Group passivation | 114 |
| 9 | Programming communication..... | 115 |
| 9.1 | Safety-related communication between F-CPU..... | 115 |
| 9.1.1 | Configuring safety-related communication via S7 connections | 115 |

| | | |
|-----------|---|------------|
| 9.1.2 | Communication via F_SENDBO/F_RCVBO, F_SENDR/F_RCVR, and F_SDS_BO/F_RDS_BO | 116 |
| 9.1.3 | Programming safety-related CPU-to-CPU communication via S7 connections | 117 |
| 9.2 | Safety-related communication between S7 F Systems and S7 Distributed Safety | 121 |
| 10 | Operator inputs with the "Secure Write Command++" function | 123 |
| 10.1 | Concept of "Secure Write Command++" | 123 |
| 10.2 | Operator functions based on "Secure Write Command++" | 125 |
| 10.3 | Programming operator functions | 126 |
| 10.3.1 | Basic procedure | 126 |
| 10.3.2 | Placement, parameter assignment and interconnection of F-blocks in the CFC | 126 |
| 10.3.2.1 | Introduction | 126 |
| 10.3.2.2 | Application case: "Change process values" with logic blocks | 128 |
| 10.3.2.3 | Application case: "Change process values" with arithmetic block | 130 |
| 10.3.2.4 | Application case: Simulating a F-channel driver | 132 |
| 10.3.2.5 | Application case: Grouped maintenance override with mutual interlock | 134 |
| 10.3.2.6 | Application case: Time-triggered maintenance override | 136 |
| 10.3.2.7 | Application case: Fail-safe acknowledgment | 138 |
| 10.3.3 | Configuring the faceplate of the operator functions | 140 |
| 10.3.4 | Integrating an operator function in an existing project | 143 |
| 10.4 | Executing operator functions | 145 |
| 10.4.1 | Requirements and general notes | 145 |
| 10.4.2 | Use of operator function "Change process value" with two operators | 146 |
| 10.4.3 | Use of operator function "Change process value" with one operator | 149 |
| 10.4.4 | Use of operator function "Maintenance Override" with two operators | 150 |
| 10.4.5 | Use of operator function "Maintenance Override" with one operator | 156 |
| 10.4.6 | Use of operator function "Fail-safe acknowledgment" with two operators | 156 |
| 10.4.7 | Use of operator function "Fail-safe acknowledgment" with one operator | 159 |
| 11 | Safety Data Write function: Changing F-parameters from the OS | 161 |
| 11.1 | Safety Data Write concept | 161 |
| 11.2 | Programming Safety Data Write | 163 |
| 11.2.1 | Basic procedure | 163 |
| 11.2.2 | Positioning, interconnecting, and assigning parameters to F-Blocks in the CFC chart | 163 |
| 11.2.3 | Examples: Safety Data Write | 165 |
| 11.2.3.1 | Example 1: F_CHG_R | 165 |
| 11.2.3.2 | Example 2: F_CHG_BO | 165 |
| 11.2.4 | Configuring the Faceplate for Safety Data Write. | 165 |
| 11.3 | Changing F-Parameters with Safety Data Write | 170 |
| 11.3.1 | Requirements and General Instructions | 170 |
| 11.3.2 | Changing an F-Parameter with Two Operators | 172 |
| 11.3.3 | Changing an F-Parameter with One Operator | 177 |
| 12 | Compiling and commissioning an S7 program | 179 |
| 12.1 | Compiling an S7 program | 179 |
| 12.2 | "Safety Program" dialog | 180 |
| 12.2.1 | "Shutdown Behavior" dialog box | 181 |
| 12.2.2 | "Logs..." button | 182 |
| 12.2.3 | "Save Reference" button | 182 |

| | | |
|-----------|--|------------|
| 12.2.4 | "Library Version" button | 182 |
| 12.2.5 | "Password for Safety Program Creation" dialog | 182 |
| 12.2.6 | "Update" button | 184 |
| 12.3 | Comparing safety programs..... | 185 |
| 12.4 | Printing project data of the safety program | 192 |
| 12.5 | Safety mode | 195 |
| 12.5.1 | Deactivating safety mode..... | 195 |
| 12.5.2 | Activating safety mode | 196 |
| 12.6 | Downloading the safety program | 198 |
| 12.6.1 | Downloading the S7 program | 199 |
| 12.7 | Testing a safety program | 200 |
| 12.7.1 | Testing with S7-PLCSIM | 200 |
| 12.8 | Modifying a safety program..... | 202 |
| 12.8.1 | Online changes in CFC test mode | 202 |
| 12.8.2 | Downloading changes..... | 203 |
| 12.8.2.1 | Changes that can be transferred by downloading changes | 205 |
| 12.8.2.2 | Changes requiring an F-Startup..... | 206 |
| 12.8.2.3 | Changes that require a cold restart or warm restart (restart) of the F-CPU | 206 |
| 12.8.2.4 | Changes that require an F-CPU STOP in a single CPU..... | 206 |
| 12.8.2.5 | Changing the time ratios or F-Monitoring times..... | 207 |
| 12.8.2.6 | Change in the safety-related communication between F-CPU's..... | 208 |
| 12.8.2.7 | Initial run and startup characteristics | 209 |
| 12.9 | Deleting the safety program..... | 210 |
| 12.10 | Acceptance test following system upgrade..... | 211 |
| 13 | System Acceptance Test | 213 |
| 13.1 | Overview of system acceptance test | 213 |
| 13.2 | Commissioning a safety program | 214 |
| 13.2.1 | Initial acceptance test of a safety program | 214 |
| 13.2.2 | Preliminary test of the configuration of the F-CPU and F-I/O (optional) | 214 |
| 13.2.3 | Backup of the STEP 7 project..... | 217 |
| 13.2.4 | Inspection of the printout..... | 217 |
| 13.2.5 | Downloading the S7 program to the F-CPU | 219 |
| 13.2.6 | Implementation of a complete function test | 219 |
| 13.3 | Acceptance test of safety program changes..... | 221 |
| 13.4 | Acceptance test of F-Block types..... | 222 |
| 14 | Operation and Maintenance | 223 |
| 14.1 | Notes on safety mode of the safety program | 223 |
| 14.2 | Replacing software and hardware components..... | 225 |
| 14.3 | F-Forcing..... | 227 |
| A | F-libraries | 229 |
| A.1 | Overview of the S7 F Systems Lib V1_3 SP2 F-library | 229 |
| A.1.1 | F-Blocks | 229 |
| A.1.2 | F-Data types..... | 230 |

| | | |
|----------|---|-----|
| A.1.3 | Block interfaces..... | 230 |
| A.1.4 | Behavior of F-Blocks with floating-point operations in the event of a number range overflow..... | 231 |
| A.1.5 | Behavior of F-Blocks in the event of safety-related faults | 231 |
| A.2 | S7 F Systems Lib V1_3 SP2 F-blocks | 232 |
| A.2.1 | Logic blocks with the BOOL data type..... | 232 |
| A.2.1.1 | Logic Blocks of the BOOL Data Type | 232 |
| A.2.1.2 | F_AND4: AND logic operation on four inputs | 232 |
| A.2.1.3 | F_OR4: OR logic operation on four inputs..... | 233 |
| A.2.1.4 | F_XOR2: XOR logic operation on two inputs | 234 |
| A.2.1.5 | F_NOT: NOT logic operation | 235 |
| A.2.1.6 | F_2OUT3: 2oo3 evaluation of inputs of data type BOOL | 235 |
| A.2.1.7 | F_XOUTY: XooY evaluation of inputs of data type BOOL..... | 236 |
| A.2.2 | F-Blocks for F-Communication between F-CPU..... | 237 |
| A.2.2.1 | F-Blocks for F-Communication between F-CPU..... | 237 |
| A.2.2.2 | F_SENDBO: Sending of 20 data elements of data type F_BOOL in a fail-safe manner to another F-CPU..... | 238 |
| A.2.2.3 | F_RCVBO: Receiving of 20 data elements of data type F_BOOL in a fail-safe manner from another F-CPU..... | 242 |
| A.2.2.4 | F_SENDR: Sending of 20 data elements of data type F_REAL in a fail-safe manner to another F-CPU..... | 246 |
| A.2.2.5 | F_RCVR: Receiving of 20 data elements of data type F_REAL in a fail-safe manner from another F-CPU..... | 249 |
| A.2.2.6 | F_SDS_BO: Sending of 32 data elements of data type F_BOOL in a fail-safe manner to another F-CPU..... | 254 |
| A.2.2.7 | F_RDS_BO: Receiving of 32 data elements of data type F_BOOL in a fail-safe manner from another F-CPU..... | 258 |
| A.2.3 | F-Blocks for comparing two input values of the same type | 262 |
| A.2.3.1 | F_CMP_R Comparator for two REAL values..... | 262 |
| A.2.3.2 | F_LIM_HL: Monitoring of upper limit violation of a REAL value..... | 263 |
| A.2.3.3 | F_LIM_LL: Monitoring of lower limit violation of a REAL value..... | 264 |
| A.2.4 | Voter blocks for inputs of data type REAL and BOOL..... | 265 |
| A.2.4.1 | F_2oo3DI: 2oo3 evaluation of inputs of data type BOOL with discrepancy analysis | 265 |
| A.2.4.2 | F_2oo3AI: 2oo3 evaluation of inputs of the REAL data type with discrepancy analysis | 267 |
| A.2.4.3 | F_1oo2AI: 1oo2 evaluation of inputs of data type REAL with discrepancy analysis | 272 |
| A.2.5 | Blocks and F-Blocks for data conversion..... | 274 |
| A.2.5.1 | Blocks and F-blocks for data conversion | 274 |
| A.2.5.2 | F_SWC_CB: Processing of a parameter of data format F-BOOL for operator input via the OS..... | 276 |
| A.2.5.3 | F_SWC_CR: Processing of a parameter of data format F-REAL for operator input via the OS | 278 |
| A.2.5.4 | F_SWC_P: Centralized control of operator input via the OS..... | 282 |
| A.2.5.5 | F_SWC_BO: Processing of a parameter of data type F_BOOL for operator input via the OS | 283 |
| A.2.5.6 | F_SWC_R: Processing of a parameter of data type F_REAL for operator input via the OS..... | 285 |
| A.2.5.7 | F_FR_FDI: Conversion from F_REAL to F_DINT | 287 |
| A.2.5.8 | F_FDI_FR: Conversion from F_DINT to F_REAL | 288 |
| A.2.5.9 | F_BO_FBO: Conversion from BOOL to F_BOOL | 288 |
| A.2.5.10 | F_R_FR: Conversion from REAL to F_REAL | 289 |
| A.2.5.11 | F_QUITES: Fail-safe acknowledgement via the ES/OS..... | 289 |
| A.2.5.12 | F_TI_FTI: Conversion from TIME to F_TIME..... | 291 |

| | | |
|----------|--|-----|
| A.2.5.13 | F_I_FI: Conversion from INT to F_INT | 292 |
| A.2.5.14 | F_FI_FR: Conversion from F_INT to F_REAL | 292 |
| A.2.5.15 | F_FR_FI: Conversion from F_REAL to F_INT | 293 |
| A.2.5.16 | F_CHG_R: Safety Data Write for F_REAL | 293 |
| A.2.5.17 | F_CHG_BO: Safety Data Write for F_BOOL | 299 |
| A.2.5.18 | F_FBO_BO: Conversion from F_BOOL to BOOL | 304 |
| A.2.5.19 | F_FR_R: Conversion from F_REAL to REAL | 304 |
| A.2.5.20 | F_FI_I: Conversion from F_INT to INT | 305 |
| A.2.5.21 | F_FTI_TI: Conversion from F_TIME to TIME | 305 |
| A.2.5.22 | SWC_CHG: Operator function for Change process values | 306 |
| A.2.5.23 | SWC_MOS: Command function for Maintenance Override | 307 |
| A.2.5.24 | SWC_QOS: Operator function for fail-safe acknowledgment | 308 |
| A.2.6 | F-Channel drivers for F-I/O | 310 |
| A.2.6.1 | F_CH_BI: F-Channel driver for inputs of data type BOOL of fail-safe DP standard slaves and fail-safe standard I/O devices | 310 |
| A.2.6.2 | F_CH_BO: F-Channel driver for outputs of data type BOOL of fail-safe DP standard slaves and fail-safe standard I/O devices | 314 |
| A.2.6.3 | F_PA_AI: Fail-safe channel driver for fail-safe "Transmitter" PA field device | 319 |
| A.2.6.4 | F_PA_DI: Fail-safe channel driver for fail-safe "Discrete Input" PA field device | 323 |
| A.2.6.5 | F_CH_DI: F-channel driver for digital inputs of F-I/O (except fail-safe DP standard slaves and fail-safe IO standard devices) | 328 |
| A.2.6.6 | F_CH_DO: F-channel driver for digital outputs of F-I/O (except fail-safe DP standard slaves and fail-safe IO standard devices) | 332 |
| A.2.6.7 | F_CH_AI: F-channel driver for analog inputs of F-I/O (except fail-safe DP standard slaves and fail-safe IO standard devices) | 336 |
| A.2.6.8 | F_CH_II: F-Channel driver for inputs of data type INT of fail-safe DP standard slaves and fail-safe standard I/O devices | 344 |
| A.2.6.9 | F_CH_IO: F-Channel driver for outputs of data type INT of fail-safe DP standard slaves and fail-safe standard I/O devices | 349 |
| A.2.6.10 | F_CH_DII: F-Channel driver for inputs of data type DINT of fail-safe DP standard slaves and fail-safe standard I/O devices | 353 |
| A.2.6.11 | F_CH_DIO: F-Channel driver for outputs of data type DINT of fail-safe DP standard slaves and fail-safe standard I/O devices | 358 |
| A.2.6.12 | F_CH_RI: F-channel driver for inputs of data type "REAL" of fail-safe DP standard slaves and fail-safe IO standard devices | 362 |
| A.2.7 | F-System blocks | 367 |
| A.2.7.1 | F_S_BO: Sending of 10 data elements of data type F_BOOL in a fail-safe manner to another F-Shutdown group. | 367 |
| A.2.7.2 | F_R_BO: Receiving of 10 data elements of data type F_BOOL in a fail-safe manner from another F-Shutdown group | 368 |
| A.2.7.3 | F_S_R: Sending of 5 data elements of data type F_REAL in a fail-safe manner to another F-Shutdown group | 369 |
| A.2.7.4 | F_R_R: Receiving of 5 data elements of data type F_REAL in a fail-safe manner from another F-Shutdown group. | 370 |
| A.2.7.5 | F_START: F-Startup identifier | 371 |
| A.2.7.6 | F_PSG_M: Marker block for F-Shutdown groups | 371 |
| A.2.8 | Flip-flop blocks | 372 |
| A.2.8.1 | F_RS_FF: RS Flip-Flop, resetting dominant | 372 |
| A.2.8.2 | F_SR_FF: SR Flip-Flop, setting dominant | 373 |
| A.2.9 | IEC pulse and counter blocks | 374 |
| A.2.9.1 | F_CTUD: Up and down counter | 374 |
| A.2.9.2 | F_TP: Timer pulse | 375 |

| | | |
|-----------|---|-----|
| A.2.9.3 | F_TON: Timer switch-on delay | 376 |
| A.2.9.4 | F_TOF: Timer switch-off delay | 378 |
| A.2.10 | Pulse blocks | 379 |
| A.2.10.1 | F_REPCYC: Clock | 379 |
| A.2.10.2 | F_ROT: Timer with on delay and hold function | 382 |
| A.2.10.3 | F_LIM_TI: Asymmetrical limiter of a TIME value | 384 |
| A.2.10.4 | F_R_TRIG: Detection of a rising edge | 385 |
| A.2.10.5 | F_F_TRIG: Detection of a falling edge | 385 |
| A.2.11 | Arithmetic blocks with the REAL data type | 386 |
| A.2.11.1 | F_ADD_R: Addition of two REAL values | 387 |
| A.2.11.2 | F_SUB_R: Subtraction of two REAL values | 387 |
| A.2.11.3 | F_MUL_R: Multiplication of two REAL values | 388 |
| A.2.11.4 | F_DIV_R: Division of two REAL values | 388 |
| A.2.11.5 | F_ABS_R: Absolute value of a REAL value | 389 |
| A.2.11.6 | F_MAX3_R: Maximum of three REAL values | 389 |
| A.2.11.7 | F_MID3_R: Mean value of three REAL values | 390 |
| A.2.11.8 | F_MIN3_R: Minimum of three REAL values | 391 |
| A.2.11.9 | F_LIM_R: Asymmetrical limiter of a REAL value | 391 |
| A.2.11.10 | F_SQRT: Square root of a REAL value | 392 |
| A.2.11.11 | F_AVE_X_R: Mean value of a maximum of nine REAL values | 393 |
| A.2.11.12 | F_SMP_AV: Sliding mean value of maximum 33 REAL values | 394 |
| A.2.11.13 | F_2oo3_R: Middle value of three REAL values with 2oo3 evaluation | 395 |
| A.2.11.14 | F_1oo2_R: 1oo2 evaluation of inputs of data type REAL | 396 |
| A.2.12 | Arithmetic blocks with the INT data type | 398 |
| A.2.12.1 | F_LIM_I: Asymmetrical limiter of an INT value | 398 |
| A.2.13 | Multiplex blocks | 399 |
| A.2.13.1 | Multiplex blocks | 399 |
| A.2.13.2 | F_MOV_R: Copy 15 values of data type REAL | 399 |
| A.2.13.3 | F_MUX2_R: Multiplexer for 2 REAL values with BOOL selection | 401 |
| A.2.13.4 | F_MUX16R: Multiplexer for 16 REAL values with INT selection | 401 |
| A.2.14 | F-Control blocks | 402 |
| A.2.14.1 | F_POLYG: F-Control block with non-linear characteristic | 402 |
| A.2.14.2 | F_INT_P: Integration function with integration and track mode | 404 |
| A.2.14.3 | F_PT1_P: First order delay | 408 |
| A.2.15 | Additional F-Blocks | 410 |
| A.2.15.1 | F_DEADTM: Monitoring of changes in F_REAL values at the same measuring point | 410 |
| A.3 | S7 F Systems Lib V1_3 SP2 F-control blocks | 414 |
| A.3.1 | F_MOVRWS: F-Control block | 415 |
| A.3.2 | F_DIAG: F-Control block | 415 |
| A.3.3 | F_CYC_CO: F-Control block "F-Cycle time monitoring" | 416 |
| A.3.4 | F_PLK: F-Control block | 417 |
| A.3.5 | F_PLK_O: F-Control block | 418 |
| A.3.6 | F_TEST: F-Control block | 419 |
| A.3.7 | F_TESTC: F-Control block | 420 |
| A.3.8 | F_TESTM: F-Control block "Deactivate Safety Mode" | 421 |
| A.3.9 | F_SHUTDOWN: F-Control block "Control of shutdown and F-Startup of the safety program" | 422 |
| A.3.10 | RTGLOGIC: F-Control block | 425 |
| A.3.11 | F_PS_12: F-Control block "F_Module_Driver" | 426 |
| A.3.12 | F_CHG_WS: F-Control block | 428 |
| A.3.13 | DB_INIT: F-Control block | 429 |
| A.3.14 | DB_RES: F-Control block | 430 |

| | | |
|----------|---|------------|
| A.3.15 | F_PS_MIX: F-Control block..... | 430 |
| A.3.16 | F_VFSTP1: F-Control block..... | 431 |
| A.3.17 | F_VFSTP2: F-Control block..... | 431 |
| A.3.18 | FORCEOFF: Deactivation of F-Force..... | 432 |
| A.4 | F-Library Failsafe Blocks (V1_2)..... | 433 |
| A.5 | Differences between the F-Libraries Failsafe Blocks (V1_x) and S7 F Systems Lib V1_3..... | 434 |
| A.5.1 | Logic blocks with the BOOL data type..... | 434 |
| A.5.2 | F-Blocks for F-Communication between F-CPU's..... | 435 |
| A.5.3 | F-Blocks for comparing two input values of the same type..... | 436 |
| A.5.4 | Voter blocks for inputs of data type REAL and BOOL..... | 438 |
| A.5.5 | Blocks and F-Blocks for data conversion..... | 438 |
| A.5.6 | F-Channel drivers for F-I/O..... | 440 |
| A.5.7 | F-System blocks..... | 443 |
| A.5.8 | Flip-flop blocks..... | 445 |
| A.5.9 | IEC pulse and counter blocks..... | 445 |
| A.5.10 | Pulse blocks..... | 446 |
| A.5.11 | Arithmetic blocks with the REAL data type..... | 447 |
| A.5.12 | Arithmetic blocks with the INT data type..... | 450 |
| A.5.13 | Multiplex blocks..... | 450 |
| A.5.14 | F-Control blocks..... | 451 |
| A.6 | Differences between the S7 F Systems Lib F-libraries..... | 456 |
| A.6.1 | Differences between the S7 F Systems Lib V1_3 SP1 and SP2 F-libraries..... | 456 |
| A.6.2 | Differences between the F-Library S7 F Systems Lib V1_3 and V1_3 SP1..... | 457 |
| A.7 | Run times, F-Monitoring times, and response times..... | 460 |
| B | Checklist..... | 461 |
| C | Requirements for virtual environments and remote access..... | 465 |
| C.1 | Summary..... | 465 |
| C.2 | Configuration and operation..... | 467 |
| C.2.1 | Virtual environments..... | 467 |
| C.2.2 | Remote Access and Control..... | 468 |
| C.3 | Examples of valid configurations in PCS 7..... | 471 |
| C.3.1 | Example 1..... | 471 |
| C.3.2 | Example 2..... | 472 |
| C.4 | Abbreviations and explanations of terms..... | 474 |
| C.5 | References..... | 475 |
| | Glossary..... | 477 |
| | Index..... | 485 |

Security information

Siemens provides products and solutions with industrial security functions that support the secure operation of plants, systems, machines and networks.

In order to protect plants, systems, machines and networks against cyber threats, it is necessary to implement – and continuously maintain – a holistic, state-of-the-art industrial security concept. Siemens' products and solutions only form one element of such a concept.

Customer is responsible to prevent unauthorized access to its plants, systems, machines and networks. Systems, machines and components should only be connected to the enterprise network or the internet if and to the extent necessary and with appropriate security measures (e.g. use of firewalls and network segmentation) in place.

Additionally, Siemens' guidance on appropriate security measures should be taken into account. For more information about industrial security, please visit:
<http://www.siemens.com/industrialsecurity>

Siemens' products and solutions undergo continuous development to make them more secure. Siemens strongly recommends to apply product updates as soon as available and to always use the latest product versions. Use of product versions that are no longer supported, and failure to apply latest updates may increase customer's exposure to cyber threats.

To stay informed about product updates, subscribe to the Siemens Industrial Security RSS Feed under:
<http://www.siemens.com/industrialsecurity>

Preface

Purpose of this documentation

The information in this manual enables you to configure and program fail-safe S7 F/FH Systems using *S7 F Systems* V6.2.

As a supplement, you need the " Safety Engineering in SIMATIC S7 (<http://support.automation.siemens.com/WW/view/en/12490443>) " system manual.

Basic Knowledge Requirements

General basic knowledge of automation engineering is needed to understand this documentation. Basic knowledge of the following is also necessary:

- Fail-safe automation systems
- S7-400 Automation Systems
- Distributed I/O systems on PROFIBUS DP / PROFINET IO
- *STEP 7* basic software, particularly:
 - Working with *SIMATIC Manager*
 - Hardware configuration with *HW Config*
 - Communication between CPUs
 - *CFC* optional software

Scope of this documentation

| | Article number | Release number and higher |
|--|---|---------------------------|
| <i>S7 F Systems</i> V6.2 optional package including authorization license V6.2 | <ul style="list-style-type: none"> • Full version: 6ES7833-1CC62-0YA5 • Upgrade version from V6.x: 6ES7833-1CC62-0YE5 | V6.2 |
| S7 F Systems RT Licence (Copy Licence) | <ul style="list-style-type: none"> • 6ES7833-1CC00-6YX0 | V5.0 |

The *S7 F Systems* optional package is used for configuring and programming S7 F/FH Systems. Integration of the F-I/Os listed below in S7 F/FH Systems is also addressed:

- ET 200S fail-safe I/O modules
- ET 200SP fail-safe I/O modules

- ET 200eco fail-safe I/O modules
- ET 200pro fail-safe I/O modules
- ET 200iSP fail-safe I/O modules
- S7-300 fail-safe signal modules (in ET 200M)
- Fail-safe DP standard slaves/IO standard devices
- Fail-safe PA field devices

What's New?

The innovations and changes in *S7 F Systems V6.2* are described below:

- New functionality
 - Operator functions for "Change process data", "Maintenance Override" and "Fail-safe acknowledgment" based on "Secure Write Command++"
- New F blocks in the F-Library:
 - F_SWC_CB: Processing of a parameter of data format F_BOOL for operator input via the OS
 - F_SWC_CR: Processing of a parameter of data format F_REAL for operator input via the OS
 - F_CH_RI: F-channel driver for inputs of data type REAL of fail-safe DP standard slaves and fail-safe IO standard devices
- New blocks in the F-Library:
 - SWC_QOS: Operator function for "Fail-safe acknowledgment"
 - SWC_CHG: Operator function for "Change process values"
- Changed functionality:
 - F_CH_AI: Integrated discrepancy analysis for redundantly configured I/O.
 - F_XOUTY: New output shows the number of inputs with signal state "1".
- Support of additional fail-safe DP standard slaves/IO standard devices
- Uniform operation philosophy of *S7 F Systems* and *PCS 7 Advanced Process Library (APL)*
- The print function allows you to select the charts to be printed.
- The dialog for comparing safety programs can be resized to make the table easier to read.
- New sample projects
- Increased complexity for the safety program password.

- When SIMATIC Logon and STEP 7 V5.5.4 HF9 or higher is used, various events from S7 F/FH Systems are logged:
 - Safety mode has been activated/deactivated.
 - Password for the safety program has been changed.
 - Access authorization for the safety program has been granted/revoked.
- With PCS 7 V8.2 or CFC V8.2 or higher, safety-related CFC charts can be opened as read-only charts when the *S7 F Systems* optional package is not installed.

Approvals

S7 F/FH Systems and the F-I/O are certified for use in safety mode for:

- Safety Integrity Level SIL3 according to IEC 61508:2010
- Performance Level (PL) e and Category 4 according to ISO 13849-1:2015 or EN ISO 13849-1:2015

Position in the information landscape

Depending on your application, you will need the following supplementary documentation when working with *S7 F/FH Systems*.

This documentation includes references to the supplementary documentation where appropriate.

| Documentation for | Brief Description of Relevant Contents |
|--|---|
| Safety Engineering in SIMATIC S7 | The "Safety Engineering in SIMATIC S7 (http://support.automation.siemens.com/WW/view/en/12490443)" system manual provides an informational overview of the use, installation and mode of operation of the S7 Distributed Safety and S7 F/FH Systems fail-safe automation systems, and describes basic properties and detailed technical information about these F-systems. |
| S7 F/FH Systems | <ul style="list-style-type: none"> • The "Automation System S7-400 Hardware and Installation (http://support.automation.siemens.com/WW/view/en/1117849)" installation manual describes the assembly and wiring of S7-400 systems. • The "Automation System S7-400H Fault-Tolerant Systems (http://support.automation.siemens.com/WW/view/en/82478488)" manual describes the CPU 41x-H central processing units and the tasks required to set up and commission an S7-400H fault-tolerant system. |
| S7 Distributed Safety | The following elements are described in the "S7 Distributed Safety - Configuring and Programming (http://support.automation.siemens.com/WW/view/en/22099875)" operating manual and online help: <ul style="list-style-type: none"> • Configuration of the F-CPU and the F-I/O • Programming of the F-CPU in F-FBD or F-LAD |
| S7-300 Automation System, ET 200M I/O Device | The "SIMATIC Automation System S7-300 ET 200M Distributed I/O Device Fail-safe Signal Modules (http://support.automation.siemens.com/WW/view/en/19026151)" manual describes the hardware of the S7-300 fail-safe signal modules (including installation, wiring and technical specifications). |

| Documentation for | Brief Description of Relevant Contents |
|----------------------------------|--|
| ET 200S Distributed I/O System | The "ET 200S Distributed I/O System - Fail-Safe Modules (http://support.automation.siemens.com/WW/view/en/12490437)" operating instructions describe the hardware of the ET 200S fail-safe modules (including installation, wiring and technical specifications). |
| ET 200SP Distributed I/O System | The Manual Collection "SIMATIC ET 200SP Manual Collection (https://support.industry.siemens.com/cs/ww/en/view/84133942)" contains all important information about the I/O system. |
| ET 200pro Distributed I/O System | The "ET 200pro Distributed I/O Device - Fail-Safe Modules (http://support.automation.siemens.com/WW/view/en/22098524)" operating instructions describe the hardware of the ET 200pro fail-safe modules (including installation, wiring and technical specifications). |
| ET 200eco Distributed I/O System | The "ET 200eco Distributed I/O Station Fail-Safe I/O Module (http://support.automation.siemens.com/WW/view/en/22099642)" manual describes the hardware of the ET 200eco fail-safe I/O module (including installation, wiring and technical specifications). |
| ET 200iSP Distributed I/O System | The "ET 200iSP Distributed I/O Device - Fail-safe Modules (http://support.automation.siemens.com/WW/view/en/47357221)" operating instructions describe the hardware of the ET 200iSP fail-safe modules (including installation, wiring and technical specifications). |
| <i>STEP 7</i> manuals | <ul style="list-style-type: none"> • The "Configuring Hardware and Communication Connections with STEP 7 V5.5 (http://support.automation.siemens.com/WW/view/en/45531110)" manual describes how to use the corresponding standard tools of <i>STEP 7</i>. • The "System software for S7 300/400 System and Standard Functions (http://support.automation.siemens.com/WW/view/en/1214574)" reference manual describes access/diagnostic functions of the distributed I/O / CPU. • The "Programming with STEP 7 V 5.5 (http://support.automation.siemens.com/WW/view/en/45531107)" manual describes the procedure for programming with <i>STEP 7</i>. • The "SIMATIC Process Control System PCS 7 CFC for SIMATIC S7 (https://support.industry.siemens.com/cs/ww/en/view/90683154)" manual/online help provides a description of programming with <i>CFC</i>. • The "Modifying the System during Operation via CiR (http://support.automation.siemens.com/WW/view/en/45531308)" manual |
| <i>STEP 7</i> Online Help | <ul style="list-style-type: none"> • Describes how to operate the standard tools of <i>STEP 7</i>. • Contains information on configuring and assigning parameters for I/Os with <i>HW Config</i>. |
| <i>PCS 7</i> | <ul style="list-style-type: none"> • The "PCS 7 manuals (www.siemens.com/pcs7-documentation)" describe operation of the <i>PCS 7</i> process control system (required when the S7 F System is integrated into a higher-level control system). <p>The documents "PCS 7 Compendium Part B - Process Safety" and "PCS 7 Compendium Part F - Industrial Security" are also available at the following link.</p> |

Guide

This documentation describes how to work with the *S7 F Systems* optional package. It includes both instructional material and reference material (description of fail-library blocks).

The following topics are addressed:

- Configuration of *S7 F Systems*
- Access protection for *S7 F Systems*
- Programming of the safety program (safety-related user program)
- Safety-related communication
- Support for the system acceptance test
- Operation and maintenance of *S7 F Systems*
- F-Libraries

Conventions

In this documentation, the terms "safety engineering" and "fail-safe engineering" are used synonymously. The same applies to the terms "fail-safe" and "F-".

When *S7 F Systems* appears in italics, it refers to the optional package for the "S7 F/FH Systems" fail-safe system.

The term "safety program" refers to the fail-safe portion of the user program and is used instead of "fail-safe user program", "F-program", etc. For purposes of contrast, the non-safety-related user program is referred to as the "standard user program".

"F-CPU" denotes a CPU with fail-safe capability. A CPU with fail-safe capability is a central processing unit that is approved for use in S7 F/FH Systems and S7 Distributed Safety.

Additional support

If you have further questions about the use of products presented in this manual, contact your local Siemens representative:

You will find information on who to contact on the Web (<http://www.siemens.com/automation/partner>).

A guide to the technical documentation for the various SIMATIC products and systems is available on the Web (<http://www.siemens.com/simatic-tech-doku-portal>).

You will find the online catalog and online ordering system on the Web (<https://mall.industry.siemens.com>).

Training center

We offer courses to help you get started with the *SIMATIC S7* automation system. Contact your regional training center or the central training center in D -90327 Nuremberg, Federal Republic of Germany.

You will find more information on the Web (<http://www.sitrain.com>).

Technical Support

To contact Technical Support for all Industry Automation products, use the Support Request Web form (<http://www.siemens.com/automation/support-request>).

You can find additional information about our Technical Support on the Web (<http://www.siemens.com/automation/service>).

Service & Support on the Internet

In addition to our paper documentation, our complete knowledge base is available to you on the Web (<http://www.siemens.com/automation/service&support>).

There, you will find the following information:

- Newsletters providing the latest information on your products
- A search engine in Service & Support for locating the documents you need
- A forum where users and experts from all over the world exchange ideas
- Your local contact partner for Industry Automation products in our Contact Partners database
- Information about on-site service, repairs, spare parts, and much more under "Repairs, spare parts, and consulting"

Important note for maintaining the operational safety of your system

Note

Systems with safety-related characteristics are subject to special operational safety requirements on the part of the operator. The supplier is also obliged to comply with special product monitoring measures. For this reason, we publish a special newsletter containing information on product developments and features that are (or could be) relevant to operation of safety-related systems. By subscribing to the relevant newsletter, you will always have the latest information and able to make changes to your system, when necessary. Just visit us on the Web (https://www.industry.siemens.com/newsletter_v4/public/AllNewsletters.aspx).

There, you can register for the following newsletters:

- S7-300/S7-300F
- S7-400/S7-400H/S7-400F/FH
- Distributed I/O
- SIMATIC Industrial Software

To receive these newsletters, select the check box "Update".

2.1 Warnings index

| Warning | Section |
|---|---------|
| Section: Product Overview | |
| S7 F/FH systems operation | 3.2 |
| Section: Installation | |
| Possible change in response time as a result of migration from Failsafe Blocks 1_2 to <i>S7 F Systems Lib V1_3 SP2</i> | 4.3 |
| Section: Configuration | |
| An F-CPU containing a safety program must have a password. | 5.3 |
| Configuring a protection level | |
| "Group diagnostics" for fail-safe F-SMs in safety mode | 5.4.2 |
| Address assignment in subnets only and in mixed configurations | |
| Identification and confirmation of the F-I/O | 5.4.3.1 |
| Devices and "F_Par_Version" parameter for a mixed configuration | 5.5 |
| Section: Access Protection | |
| Limiting access using the ES | 6.2 |
| Transferring the safety program to multiple F-CPU's | |
| Password protection | |
| Limiting access using the ES | 6.3 |
| Passwords must be unique | |
| Section: Programming | |
| Default setting of the maximum MAX_CYC | 7.2.3 |
| Do not change values created during compilation | 7.2.4 |
| The call interval of cyclic interrupt OB 3x is monitored for the maximum value | |
| Compression changes the signature | 7.2.5 |
| Optimization of the run sequence in CFC | 7.2.7 |
| Entries for F-Blocks in the symbol table must not be changed | 7.3.1 |
| Illegal changes to input parameters of F-Blocks can cause a shutdown of the safety program and its outputs | 7.3.2 |
| Do not change automatically inserted F-Control blocks. | 7.4 |
| Saved error information is lost during an F-Startup | 7.5 |
| Outputs of F-blocks always use the predefined initial values | 7.7.2 |
| Validity check | 7.9.2 |
| The two acknowledgment steps must not be triggered with a single operation. | 7.10 |
| If your OS can access multiple F-CPU's | |
| Section: F-I/O access | |
| For F-I/O with inputs, the fail-safe value 0 provided at the F-channel driver must be further processed for (digital) channels of data type BOOL in the safety program. | 8.3 |
| Section: Programming communication | |
| CPU-CPU communication and public networks | 9.1.1 |
| The value for the respective address relationship | 9.1.3 |

| Warning | Section |
|---|----------------------|
| It can only be ensured (from a fail-safe standpoint) that a signal level to be transmitted will be detected on the sender side and transmitted to the receiver if the signal level is present for at least as long as the assigned F-monitoring time (TIMEOUT). | |
| If the F-CPU with the associated F_SENDBO/F_SDS_BO/F_SENDR is in deactivated safety mode, you can no longer assume that the data received from this F-CPU were generated safely. | |
| The S7 program must be recompiled if the S7 connections for communication between F-CPU's have been changed. | |
| Section: Operator inputs with the "Secure Write Command++" function | |
| The "Secure Write Command++" functionality allows changes to the safety program to be made during RUN mode | 10.3.2.1 |
| Warnings in the descriptions of the F-blocks | 10.3.2.2 to 10.3.2.7 |
| Restoration of edited faceplates | 10.3.3 |
| Initiator and confirmer must not accept an invalid value | 10.4.1 |
| Technological assignment must be appropriate for the environment | |
| Transaction for changing an F-Parameter | |
| Section: Safety Data Write function | |
| Warnings in the descriptions of the F-blocks | 11.2.2 |
| Static values of the SAFE_ID1 and SAFE_ID2 attributes | 11.2.4 |
| Initiator and confirmer must not accept an invalid value | 11.3.1 |
| Technological assignment must be appropriate for the environment | |
| Transaction for changing an F-Parameter | |
| Section: Compiling and commissioning an S7 program | |
| Deactivating safety mode | 12.5.1 |
| Do not copy F-Blocks with <i>SIMATIC Manager</i> | 12.6 |
| Safety program on a memory card | 12.6.1 |
| If multiple F-CPU's can be reached from an ES via a network (e.g. MPI) | |
| Shutdown of the safety program following a change to the fail-safe outputs | 12.7 |
| A simulation is no substitute for a function test! | 12.7.1 |
| Changing the collective signature for changes in CFC test mode | 12.8.1 |
| Do not change values created during compilation | |
| Download operation aborted | 12.8.2 |
| Moving F-Blocks or F-Runtime groups | |
| Modifying the safety program in RUN mode | |
| Section: Acceptance of the system | |
| Address assignment in subnets only and in mixed configurations | 13.2.1 |
| Section: Operation and maintenance | |
| If you operate simulation devices or simulation programs | 14.1 |
| Switching from STOP to RUN from the ES | |
| STOP status initiated with SFC 46 "STP" | |
| Two F-CPU's not simultaneously as master system | |
| Using the "F-Forcing" function | 14.3 |
| Section: F-Libraries | |

| Warning | Section |
|---|-----------------------|
| Values of PAR_ID and COMPLEM must not be changed | A.1.2 |
| Value for the respective address relationship | A.2.2.2 |
| Detecting and transmitting a signal level | |
| Value for the respective address relationship | A.2.2.3 |
| Measure and transfer signal level | |
| User acknowledgement is always required for communication errors | A.2.2.4 |
| Value for the relevance address reference | |
| Detecting and transmitting a signal level | A.2.2.5 |
| Value for the respective address relationship | |
| Measure and transfer signal level | A.2.2.6 |
| User acknowledgement is always required for communication errors | |
| Value for the relevance address reference | A.2.2.7 |
| Detecting and transmitting a signal level | |
| Value for the respective address relationship | A.2.4.1 to A.2.4.3 |
| Measure and transfer signal level | |
| User acknowledgement is always required for communication errors | A.2.5.1 |
| Fail-safe user times | |
| Validity check | A.2.5.2 |
| The "Secure Write Command++" functionality allows changes to the safety program to be made during RUN mode. | |
| The CHANGED input cannot be evaluated in the safety program. | |
| Interconnection of the CS_VAL input is not permitted. | |
| F-Startup | A.2.5.3 |
| The "Secure Write Command++" functionality allows changes to the safety program to be made during RUN mode. | |
| The CHANGED input cannot be evaluated in the safety program. | |
| The CS_VAL, MIN, MAX and MAXDELTA inputs must not be interconnected. | |
| F-startup | A.2.5.4 |
| The "Secure Write Command++" functionality allows changes to the safety program to be made during RUN mode. | |
| The "Secure Write Command++" functionality allows changes to the safety program to be made during RUN mode. | A.2.5.5 |
| Interconnection of the CS_VAL input is not permitted. | |
| F-startup | A.2.5.6 |
| The "Secure Write Command++" functionality allows changes to the safety program to be made during RUN mode. | |
| The CS_VAL, MIN and MAX inputs must not be interconnected. | |
| F-Startup | A.2.5.11 |
| Reintegration through user acknowledgement with F_QUITES | A.2.5.16 |
| The "Safety Data Write" functionality allows changes to the safety program to be made during RUN mode. | |
| The CHANGED output cannot be evaluated in the safety program | |
| The MIN, MAX, and MAXDELTA inputs must not be interconnected. | |

| Warning | Section |
|--|----------------------|
| Parameters SAFE_ID1 and SAFE_ID2 | |
| F-Startup | |
| The "Safety Data Write" functionality allows changes to the safety program to be made during RUN mode. | A.2.5.17 |
| The CHANGED output cannot be evaluated in the safety program | |
| Parameters SAFE_ID1 and SAFE_ID2 | |
| F-Startup | |
| Assignment of input ACK_NEC = 0 is only allowed if an automatic reintegration is permissible under safety aspects for the process. | A.2.6.1 |
| Startup protection for short-term power failure of the fail-safe DP standard slave/IO standard device | |
| Assignment of input ACK_NEC = 0 is only allowed if an automatic reintegration is permissible under safety aspects for the process. | A.2.6.2 to A.2.6.4 |
| Startup protection for short-term power failure of the fail-safe PA field device | |
| Assignment of input ACK_NEC = 0 is only allowed if an automatic reintegration is permissible under safety aspects for the process. | A.2.6.5 to A.2.6.7 |
| Startup protection for short-term power failure of the F-I/O | |
| Assignment of input ACK_NEC = 0 is only allowed if an automatic reintegration is permissible under safety aspects for the process. | A.2.6.8 to A.2.6.12 |
| Startup protection for short-term power failure of the fail-safe DP standard slave/IO standard device | |
| Fail-safe user times | A.2.9.2 to A.2.9.4 |
| Fail-safe user times | A.2.10.1 to A.2.10.2 |
| F-Startup | A.2.13.2 |
| Safety note: Do not change automatically inserted F-control blocks | A.3 to A.3.3 |
| Default setting of the maximum MAX_CYC | A.3.3 |
| Safety note: Do not change automatically inserted F-control blocks | A.3.4 to A.3.18 |
| Section: Requirements for virtual environments and remote access | |
| Use of virtual environments in ES/OS | C.2.1 |
| Remote access from higher-level control room and Engineering Center | C.2.2 |
| The "S7 F Systems HMI" and "Safety Matrix Viewer" functionality makes changes in the safety program during RUN mode. | |

Product Overview

3.1 Overview

S7 F/FH Systems fail-safe systems

The fail-safe automation systems ("F-Systems") S7 F/FH Systems are used in systems with stringent safety requirements. The objective of S7 F/FH systems is to control processes with an immediately achievable safe state. In other words, F-Systems control processes in which an immediate shutdown does not endanger people or the environment.

The *S7 F Systems* optional package comprises the following components:

- *S7 F Systems*
- *S7 F Systems HMI*
- *S7 F Systems Lib V1_3*
- *S7 F Device Integration Pack*
- *S7 F Configuration Pack*
- *Automation License Manager*

The related version designations can be found in the readme file.

Achievable safety requirements

With S7 F/FH Systems, you achieve the following safety requirements:

- Safety Integrity Level SIL3 according to IEC 61508:2010
- Performance Level (PL) e and Category 4 according to ISO 13849-1:2015 or EN ISO 13849-1:2015

The principle of safety functions in S7 F/FH Systems

Functional safety is implemented principally through safety functions in the software. Safety functions are performed by S7 F/FH Systems whenever a dangerous event occurs:

- To place the system in a safe state

or

- To keep the system in a safe state

Safety functions are contained mainly in the following components:

- In the safety-related user program (safety program) in the fail-safe CPU (F-CPU)
- In the fail-safe inputs and outputs (F-I/O)

The F-I/O ensures safe processing of field information (such as temperature and level monitoring). They have all of the required hardware and software components for safe

processing, in accordance with the required safety class. You only have to program the user safety function. The safety function for the process can be provided through a user safety function or a fault reaction function. In the event of a fault, if the F-system can no longer execute its actual user safety function, it executes the fault reaction function. For more information, refer to section "F-STOP (Page 93)".

Example of user safety functions and fault reaction functions

In the event of overpressure, the F-system opens a valve (user safety function). In the event of a dangerous fault in the F-CPU, all outputs are switched off (fault reaction function). The valve is opened and the other actuators also achieve a safe state. If the F-system is intact, only the valve would be opened.

Fail-safety and availability

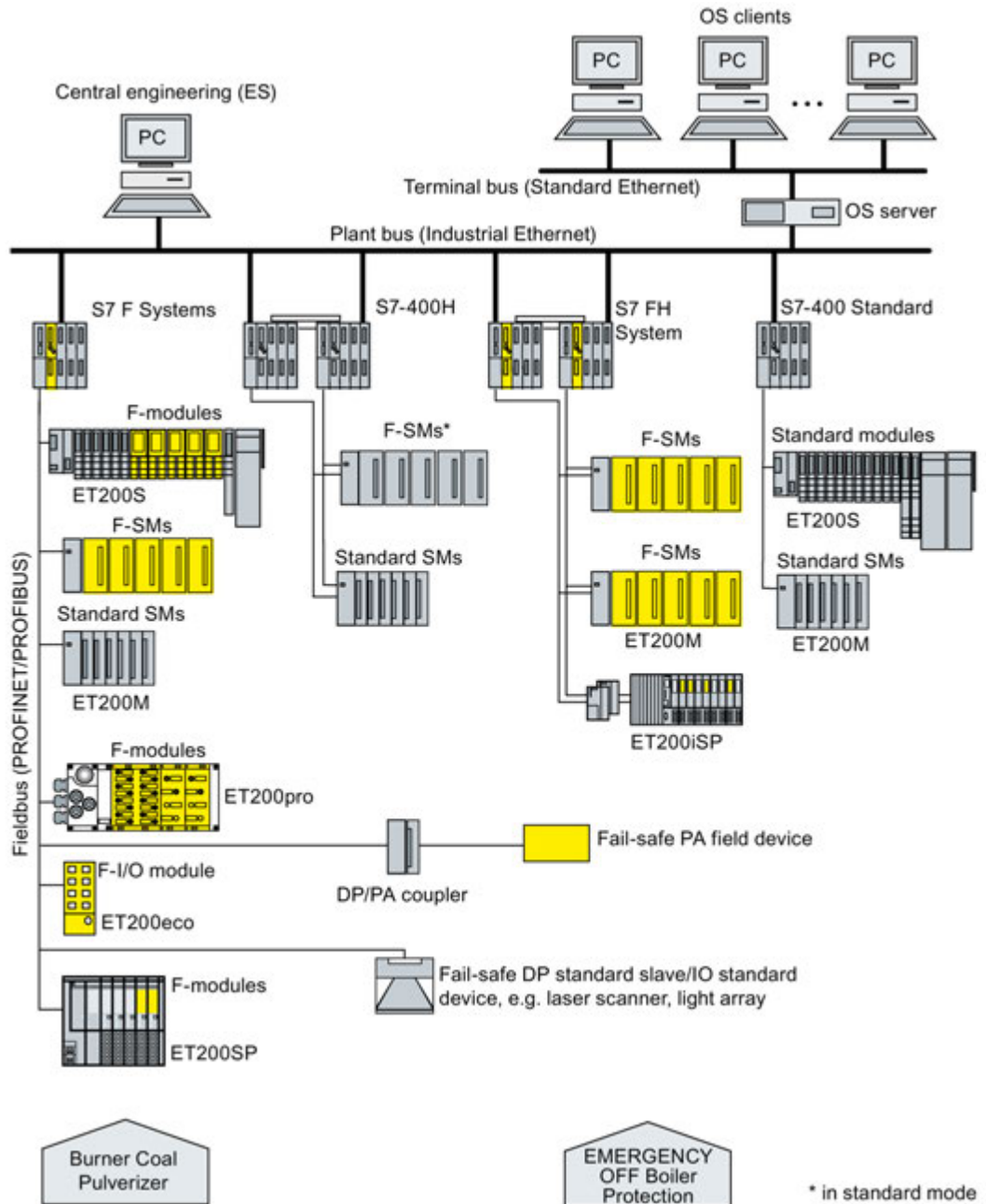
To increase availability of the automation system and, thus, to prevent process failures due to faults in the F-System, you can optionally equip fail-safe systems with a fault-tolerant feature. You achieve this increased availability through component redundancy:

- Power supply
- Central processing unit
- Communication
- F-I/O

With fail-safe, high-availability S7 F/FH Systems, you can resume production without harming people or the environment.

Use in process engineering

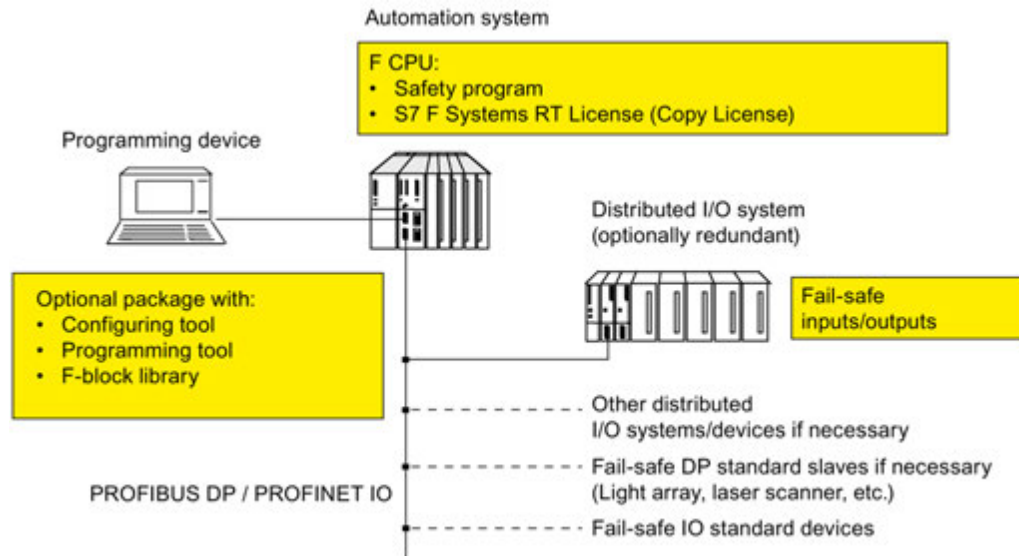
The figure below shows you the possible ways of integrating S7 F/FH Systems into your process automation system with PCS 7.



3.2 Hardware and software components

Hardware and software components of S7 F/FH systems

The following figure gives you an overview of hardware and software components you need to set up and operate S7 F/FH systems.



Hardware components

The hardware components of S7 F/FH systems include:

- F-CPU (CPU 412-5H, CPU 414-5H, CPU 416-5H, CPU 417-5H or CPU 410-5H)
- Fail-safe inputs/outputs (F-I/O), for example:
 - S7-300 fail-safe signal modules in ET 200M (distributed configuration)
 - Fail-safe power and electronic modules in ET 200S
 - Fail-safe power and I/O modules in ET 200SP
 - ET 200eco fail-safe I/O modules
 - ET 200pro fail-safe I/O modules
 - ET 200iSP fail-safe modules
 - Fail-safe DP standard slaves
 - Fail-safe IO standard devices
 - Fail-safe PA field devices.

You can expand the configuration with standard I/O.

Note**F-I/O for PCS 7**

For information on the F-I/O released for PCS 7, please refer to the "Process Control System PCS 7 Released Modules" manual for the respective PCS 7 version.

Software components
 **WARNING**
S7 F/FH systems operation

You may only operate S7 F/FH systems in the approved system environments.

Operation in a virtual environment or remote access are permitted under the conditions listed in section "Requirements for virtual environments and remote access (Page 465)".

The software components of S7 F/FH systems include:

- The *S7 F Systems* optional package on the ES for configuring and programming the F-system.
- The safety program in the F-CPU
- *S7 F Systems HMI* for displaying the F-operator control blocks on the OS

You also need the *STEP 7* basic software and *CFC* optional software on the ES for configuration and programming.

S7 F Systems optional package

This documentation describes *S7 F Systems*. *S7 F Systems* is the configuration and programming software for S7 F/FH systems. With *S7 F Systems*, you obtain:

- Support in the configuration of the F-I/O in *STEP 7* with *HW Config*
- Support in creating the safety program and integrating error detection functions in the safety program
- The F-library with F-blocks that you can use in your safety program.
- *S7 F Systems* also offers functions for comparing safety programs and to assist you in the acceptance of your plant.
- Support for operator control of fail-safe parameters of a *PCS 7 OS* during operation (Secure Write Command/Safety Data Write).
- Support for safety-related changes of F-parameters in the safety program of the F-CPU from a *PCS 7 OS* (Maintenance Override).
- Support for fail-safe acknowledgment from a *PCS 7 OS* (with SWC_QOS/F_QUITES).
- Support for operation and maintenance with F-forcing.

Safety program

You can create a safety program with the *CFC Editor* in *STEP 7* from the F-blocks that are included in an F-library with the *S7 F Systems* optional package.

When you compile the S7 program, safety checks are automatically performed and additional F-blocks for fault detection and fault response are installed. This ensures that failures and errors are detected and appropriate reactions triggered. This keeps the F-system in a safe state or brings the system to a safe state.

The S7 program in the CPU is comprised of fail-safe (safety program) and non-fail-safe (standard user program) components.

Data can be exchanged between safety and standard user programs in the F-CPU using special F-blocks for data conversion.

Installing

4.1 Installing the S7 F Systems optional package

Software requirements

You must install the following software packages in order to operate *S7 F Systems* V6.2:

- On the ES
 - *STEP 7* V5.5 SP3 or higher
 - *CFC* V8.0 SP4 or higher
 - Optional: *PCS 7* V8.0 SP2 or higher
- On the OS (for *S7 F Systems* HMI)
 - *PCS 7* V8.0 SP2 or higher
- For off-line testing
 - *S7 PLCSIM* V5.4 or higher

Available installation units

S7 F Systems comprises the following installation units:

The related version designations can be found in the readme file.

- Engineering:
 - *S7 F Systems*
 - *S7 F Systems HMI*
 - *S7 F Systems Lib* V1_3
 - *S7 F Device Integration Pack*
 - *S7 F Configuration Pack*
 - *Automation License Manager*
- Runtime:
 - *S7 F Systems HMI*

Observe the installation notes in section 3 of the "*S7 F Configuration Pack - Readme*" file for *S7 F Configuration Pack*.

4.1 Installing the S7 F Systems optional package

Reading readme files

You will find important information about the supplied software in the readme files "S7 F Systems - readme", "S7 F Configuration Pack - readme," and "S7 F Systems HMI - readme". You can display these files at the end of the corresponding setup program. At a later point, you can open the readme file by selecting **Siemens Automation > SIMATIC > Information** in the Windows Start menu.

Installing S7 F Systems

1. Start your ES/workstation. Ensure that no *STEP 7* applications are open.
2. Insert the optional package product CD.
3. Initiate the SETUP.EXE program on the CD.
4. Follow the setup program instructions.

Starting S7 F Systems

The *S7 F Systems* optional package does not contain any applications that you must start specifically. Support for the configuration and programming of F-Systems is integrated into:

- *SIMATIC Manager*
- *HW Config*
- *CFC Editor*
- *PCS 7 OS*

Displaying integrated Help

Context-sensitive Help is provided for the dialogs of the optional package. You can access this Help at every stage of configuring and programming using the F1 key or the "Help" button. For advanced help, select the menu command **Help > Contents > Calling Help on Optional Packages > Help on STEP 7 > S7 F/FH Systems - Working with F Systems**.

License key (usage authorization)

A license key is required for the *S7 F Systems* optional package. This license key is installed in the same way as for *STEP 7* and the optional packages. For information on installing and working with license keys, refer to the readme file and the *STEP 7* basic Help.

S7 F Systems RT License (Copy License)

The S7 F Systems RT license (copy license) allows you to use a CPU as an F-CPU (for example, to run a safety program on it).

4.2 Uninstalling the S7 F Systems optional package

Removing *S7 F Systems*

The *S7 F Systems* optional package comprises the following components:

- *S7 F Systems*
- *S7 F Systems HMI*
- *S7 F Systems Lib V1_3*
- *S7 F Device Integration Pack*
- *S7 F Configuration Pack*

These components can be individually removed. Use the normal procedure in Windows for removing software:

1. You can open the "Control Panel" from the Windows Start menu. In the Control Panel, open the dialog for installing software, e.g. in Windows 7 via the "Programs > Programs and Features" icons.
2. Select the appropriate entry in the list of installed software. Select "Uninstall" from the shortcut menu to uninstall the software.

4.3 Migration to S7 F Systems V6.2

Introduction

Before you migrate from an existing project to *S7 F Systems* V6.2, read the following section carefully. This section contains the following important information:

- Basic information on migration
- Possible consequences of a migration
- User scenarios for a migration

Note

S7 F Systems V6.2 supports more F-I/O than *PCS 7* if necessary, consult the documentation for *PCS 7*.

With these F-I/O, however, only the processing with *S7 F Systems* and not the diagnostic functionality of *PCS 7* is generated during compilation. For this reason, the message "Module is not supported" appears on the "Module drivers" tab when compiling.

Note

If you want to use the new functions, an update of *S7 F Systems* to V6.2 requires a simultaneous update of the *S7 F Systems Lib* to V1_3 SP2.

Note

Specific notes on compatibility

- *S7 F Systems* V6.2 is compatible with *Safety Matrix* V6.1 SP1 and higher.
 - *S7 F Systems HMI* V6.2 is compatible with *S7 F Systems Lib* V1_3 SP1.
-

Migrating to *S7 F Systems* V6.2

Note

Proceed according to the scenarios described here when migrating. Do not use the "Update block types" function even for multiprojects. Proceed as described in section "Updating a multiproject master data library (Page 43)" to update a multiproject master data library.

Note

Upgrade of communication blocks

When F-communication blocks from *S7 F Systems Lib* V1_3 SP2 are used, please note the new input parameter "COMMVER_USED" of the F-receive blocks F_RCVBO, F_RCVR and F_RDS_R. The description of the new input parameter can be found in the associated block descriptions in section "F-Blocks for F-Communication between F-CPU's (Page 237)".

Before upgrading a specific project to *S7 F Systems V6.2*, you must choose one of these two variants:

| Variant | Consequences | |
|---------------------------------|--|--|
| | Advantages | Disadvantages |
| Without update of the F-library | <ul style="list-style-type: none"> • Safety program is not changed • New acceptance may not be necessary | <ul style="list-style-type: none"> • No new F-blocks for new functionality • No support of new F-I/O released in the future |
| With update of the F-library | <ul style="list-style-type: none"> • All new functions can be used • F-I/O released in the future is supported | <ul style="list-style-type: none"> • The safety program is changed by the migration • A full download (with STOP) must be used when downloading the program to the F-CPU |

Note

"S7 F Systems Lib V1_3" library

In the descriptions and user scenarios below, "S7 F Systems Lib V1_3" is used to designate the library. In doing so, "S7 F Systems Lib V1_3" is regarded as a representative of the installed version of the library.

You can find the installed version in the Windows Start menu in subdirectory "Siemens Automation > SIMATIC > Installed software".

Migration without update of the F-library

A migration without update is simply a software update on your ES:

| Migration from ... | to <i>S7 F Systems V6.2</i> |
|--------------------------------------|-----------------------------|
| <i>S7 F Systems V6.0 to V6.1 SP2</i> | User scenario 1 (Page 34) |

Requirements:

- *S7 F Systems* in a version V6.0 to V6.1 SP2 is installed on your ES.
- *S7 F Systems Lib V1_3* or higher is used as the F-library.

Note

If you use an older version of *S7 F Systems* than V6.0, you must check whether the operating system of the ES meets the minimum requirements of *S7 F Systems V6.2*. If required, you must upgrade your operating system on the ES before installing *F Systems V6.2*.

Migration with update of the F-library

The steps you must perform depend on which F-library is used in your S7 program. Identify the scenario matching your situation from the following table:

| Migration from ... | to <i>S7 F Systems Lib V1_3 SP2</i> |
|--|-------------------------------------|
| <i>Failsafe Blocks (V1_1)</i> | User scenario 2 (Page 35) |
| <i>Failsafe Blocks (V1_2)</i> | User scenario 3 (Page 39) |
| <i>S7 F Systems Lib V1_3 or higher</i> | User scenario 4 (Page 42) |

Note

Refer also to sections "Differences between the F-Libraries Failsafe Blocks (V1_x) and S7 F Systems Lib V1_3 (Page 434)" and "Differences between the S7 F Systems Lib F-libraries (Page 456)".



WARNING

Possible change in response time as a result of migration from Failsafe Blocks 1_2 to *S7 F Systems Lib V1_3 SP2*

The maximum response time may change as a result of migration to *S7 F Systems Lib V1_3 SP2*. Use the Excel file S7FTIMEA.XLS to calculate the new maximum response time of your S7 F/FH System. For more information, refer to section "Run times, F-Monitoring times, and response times (Page 460)".

Proceed as described in the user scenarios relevant for you.

The following sections give you a description of the user scenarios.

4.3.1 User scenario 1

Objective

Simple software update from *S7 F Systems* in a version V6.0 to V6.1 SP2 to *S7 F Systems V6.2* without a program change.

Introduction

This user scenario helps you when migrating from *S7 F Systems* in a version V6.0 to V6.1 SP2 when you want to retain compatibility with your previous version.

Requirement

Your S7 program must be compiled, downloaded and executable for the original *S7 F Systems Lib V1_3*. Ensure this through a printout of the safety program and an online comparison.

Consequences

- The safety program is not changed.
- The collective signature is not changed.

Procedure

1. Install *S7 F Systems V6.2*.
2. Prior to the initial compilation, save the current version of your safety program as a reference ("Save reference" in the "Safety program" dialog) so that it will be available for future comparisons.
3. You can now compile your S7 program again.

4.3.2 User scenario 2

Objective

Upgrade of your S7 program with *Failsafe Blocks (V1_1)* to *S7 F Systems Lib V1_3 SP2*.

Introduction

This user scenario helps you when migrating your safety program by upgrading blocks of the *Failsafe Blocks (V1_1)* F-Library to blocks of the *S7 F Systems Lib V1_3 SP2* F-Library. You can then use the new functions of the *S7 F Systems Lib V1_3 SP2* F-Library.

When you migrate from the *Failsafe Blocks (V1_1)* F-Library to *S7 F Systems Lib V1_3 SP2*, the F-FBs in your safety program are overwritten by F-blocks with other block signatures. This means that the collective signature will change.

In Version V5.1 of *S7 F Systems*, you had to place the F_CYC_CO F block manually. When migrating to *S7 F Systems Lib V1_3 SP2*, this F-block is automatically moved to a system runtime group.

The shutdown logic is adapted automatically during compilation. The shutdown logic has interfaces with each F-runtime group.

Note

Different behavior for safety-related faults

In *S7 F Systems V6.2* with *S7 F Systems Lib V1_3 SP2*, F-blocks do not initiate a CPU STOP when a safety-related error (in the safety data format, for example) is detected. Instead, the shutdown logic shuts down either the F-Shutdown group affected by the error or the entire safety program (F-STOP).

You can configure the shutdown logic accordingly as:

- Partial shutdown
Only the affected F-Shutdown group is shut down.
- Full shutdown
The entire safety program is shut down.

For more information, refer to sections "F-Shutdown groups (Page 84) " and "F-STOP (Page 93)".

Note

Different behavior for floating point operations

With the *Failsafe Blocks (V1_1)* F-Library, a CPU-STOP was initiated when a floating point operation resulted in an overflow (\pm infinity) or a denormalized or invalid (NaN) floating point number, or when an invalid floating point number (NaN) was already present as an address.

Starting with *S7 F Systems Lib V1_3*, these events no longer cause a CPU-STOP. The results "Overflow (\pm infinity)," "Denormalized floating point number," or "Invalid floating point number (NaN)" are either:

- Output at the output and can be further processed by subsequent F blocks
- or*
- Signaled at special outputs. A fail-safe value is output, if necessary.

If the floating-point operation yields an invalid floating point number (NaN) without the existence of a previous invalid floating point number (NaN) as an address, the following diagnostics event is recorded in the diagnostics buffer of the F-CPU:

- "Safety program: invalid REAL number in DB" (Event ID 16#75D9)

You can use this diagnostics buffer entry to identify the F block with the invalid floating-point number (NaN).

Also refer to the documentation for F-blocks in appendix "F-libraries (Page 229)".

If you cannot rule out the occurrence of these events in your safety program, you must decide independent of your application whether you must react to them in your safety program. With the F block F_LIM_R, you can check the result of a floating-point operation for overflow (\pm infinity) and invalid floating-point number (NaN).

Note

With the *Failsafe Blocks* (V1_1) F-Library, a discrepancy analysis was not performed with redundant F-I/O and "2-channel equivalent" type of sensor interconnection in the F-Module driver, even if a discrepancy time greater than or less than (<>) 0 ms (10 ms by default) was configured in the "Redundancy" tab in *HW Config*.

With *S7 F Systems Lib V1_3* and *S7 F Configuration Pack V5.5 SP3* and higher, a discrepancy analysis is always performed for a discrepancy time greater than or less than (<>) 0 ms.

If you want to shut down the discrepancy analysis, configure a discrepancy time equal to (=) 0 ms in the "Redundancy" tab in *HW Config*.

Requirement

If F-block types are used in your project, you must re-create these with *S7 F Systems Lib V1_3 SP2* beforehand. To do so, follow the procedure in section "Updating custom F-block types (Page 43)".

Consequences

- The collective signature is changed
- A complete download with CPU-STOP is required

Procedure

1. Install *S7 F Systems V6.2* with *S7 F Systems Lib V1_3 SP2*.
2. Prior to the initial compilation, save the current version of your safety program as a reference ("Save reference" in the "Safety program" dialog) so that it will be available for future comparisons.
3. In the Safety Program dialog, select the *S7 F Systems Lib V1_3* F-Library.
To do so, click the "Library Version" button in the "Edit safety program" dialog.
4. In the S7 program, update the existing F-block types. See section "Updating custom F-block types (Page 43)" for more on this.
5. Update all block types in the *CFC Editor* by selecting **Options > Block Types** and clicking "New Version".
6. In the *CFC Editor* under **Options > Block Types**, click "Clean Up".
7. Compile your hardware configuration.
8. Compile your S7 program.

Additional measures for F-Module drivers

For migration to *S7 F Systems Lib V1_3 SP2*, interconnections of the following outputs of the F-module drivers may require special handling:

- PROFIsafe1
- PROFIsafe2
- DIAG_1
- DIAG_2

Proceed as follows when migrating to *S7 F Systems Lib V1_3 SP2*:

1. Before performing the migration, document the interconnection of the PROFIsafe1 and DIAG_1 outputs, along with the value of the LADDR input
2. For redundant F-I/O, also document the interconnection of the PROFIsafe2 and DIAG_2 outputs before performing the migration, along with the value of the LADDR_R input
3. Perform the migration to *S7 F Systems Lib V1_3 SP2*.
4. Interconnect the documented interconnections at the PROFIsafe1 and DIAG_1 outputs to the new F-Module driver F_PS_12, whose value at the LADDR input matches the documented LADDR
5. For redundant F-I/O, interconnect the documented interconnections at the PROFIsafe2 and DIAG_2 outputs to the new F-Module driver F_PS_12, whose value at the LADDR input matches the documented LADDR_R

Table 4- 1 Non-redundant F-I/O

| <i>Failsafe Blocks (V1_1)</i> | <i>S7 F Systems Lib V1_3 SP2</i> |
|---|--|
| Interconnection at the original F-Module driver: | Interconnection at the F-Module driver F_PS_12: |
| PROFIsafe1 | PROFIsafe |
| DIAG_1 | DIAG |
| LADDR | LADDR |

Table 4- 2 Redundant F-I/O

| <i>Failsafe Blocks (V1_1)</i> | <i>S7 F Systems Lib V1_3 SP2</i> |
|---|--|
| Redundant interconnection at the original F-Module driver: | Interconnection at the 1st F-module driver F_PS_12: |
| PROFIsafe1 | PROFIsafe |
| DIAG_1 | DIAG |
| LADDR | LADDR |
| | Interconnection at the 2nd F-module driver F_PS_12: |
| PROFIsafe2 | PROFIsafe |
| DIAG_2 | DIAG |
| LADDR_R | LADDR |

Additional measures for redundant fail-safe digital input modules SM 326; DI 8 X NAMUR and SM 326; DI 24 X DC 24 V

With the redundant fail-safe digital input modules SM 326; DI 8 X NAMUR and SM 326; DI 24 X DC 24 V, information about detected discrepancy errors is provided at the DIAG_1 and DIAG_2 outputs of the F_M_DI8 and F_M_DI24 F block drivers when the *Failsafe Blocks* (V1_1) F-Library is used.

Starting with *S7 F Systems Lib* V1_3 SP1, discrepancy error information is output at the DISCF and DISCF_R outputs of the F-Channel driver F_CH_DI.

If you are using logic that evaluates this information, modify it accordingly.

4.3.3 User scenario 3

Objective

Upgrade of your S7 program with *Failsafe Blocks* (V1_2) to *S7 F Systems Lib* V1_3 SP2.

Introduction

This user scenario helps you when migrating your safety program by upgrading blocks of the *Failsafe Blocks* (V1_2) F-Library to blocks of the *S7 F Systems Lib* V1_3 SP2 F-Library, enabling you to use the new functions of the F-library.

When you migrate from the *Failsafe Blocks* (V1_2) F-Library to *S7 F Systems Lib* V1_3 SP2, the F-FBs in your safety program are overwritten by F-blocks with other block signatures. This means that the collective signature will change.

Note

With the *Failsafe Blocks* (V1_2) F-Library, a discrepancy analysis was not performed with redundant F-I/O and "2-channel equivalent" type of sensor interconnection in the F-module driver, even if a discrepancy time greater than or less than (<>) 0 ms (10 ms by default) was configured in the "Redundancy" tab in *HW Config*.

In *S7 F Systems Lib* V1_3 and *S7 F Configuration Pack* V5.5 SP3 and higher, a discrepancy analysis is always performed for a discrepancy time greater than or less than (<>) 0 ms.

If you want to deactivate the discrepancy analysis, configure a discrepancy time equal to (=) 0 ms in the "Redundancy" tab in *HW Config*.

You can find advanced information on upgrading S7 F Systems Failsafe Blocks Lib V1.2 to V1.3 on the Support pages under FAQ "What must be observed when upgrading Failsafe Blocks (V1_2) to S7 F Systems Lib V1_3?" (<http://support.automation.siemens.com/WW/view/en/30375362>).

Requirements

- If F-block types are used in your project, you must recreate them with *S7 F Systems Lib V1_3 SP2* beforehand. To do this, follow the procedure outlined in section "Updating custom F-block types (Page 43)".
- If you are using project-wide unique assignment of alarm numbers (old method), please follow these guidelines to ensure you have available the full scope of the alarms for the state of the safety program:
 - Disable the alarm configuration for F_SHUTDN before upgrading to *S7 F Systems Lib V1_3*.

Consequences

- The collective signature is changed
- A complete download with CPU-STOP is required

Procedure

1. Install *S7 F Systems V6.2* with *S7 F Systems Lib V1_3 SP2*.
2. Prior to the initial compilation, save the current version of your safety program as a reference ("Save reference" in the "Safety program" dialog) so that it will be available for future comparisons.
3. In the Safety Program dialog, select the *S7 F Systems Lib V1_3 F-Library*.
To do so, click the "Library Version" button in the "Edit safety program" dialog.
4. In the S7 program, update the existing F-block types. See section "Updating custom F-block types (Page 43)" for more on this.
5. Update all block types in the *CFC Editor* by selecting **Options > Block Types** and clicking "New Version".
6. In the *CFC Editor* under **Options > Block Types**, click "Clean Up".
7. Compile your hardware configuration.
8. Compile your S7 program.

Additional measures if your project contains the F blocks F_1oo2_R or F_2oo3_R

The F blocks F_1oo2_R and F_2oo3_R have their own DELTA input. This input has the data type F_REAL in *S7 F Systems Lib V1_3 SP2*. Until *Failsafe Blocks (V1_2)*, the DELTA input had the data type REAL.

Proceed as follows when migrating to *S7 F Systems Lib V1_3 SP2*:

1. Before upgrading, document the parameter assignments and interconnections to this input
2. Perform the migration to *S7 F Systems Lib V1_3 SP2*.
3. Reintroduce the documented parameter assignments and interconnections into your project, using the F_R_FR converter and a validity check, if necessary For information

about the validity check, refer to section "Programming data exchange from the standard user program to the safety program (Page 104)".

Additional measures for F-Module drivers

For migration to *S7 F Systems Lib V1_3 SP2*, interconnections of the following outputs of the F-module drivers may require special handling:

- PROFIsafe1
- PROFIsafe2
- DIAG_1
- DIAG_2

Proceed as follows when migrating to *S7 F Systems Lib V1_3 SP2*:

1. Before performing the migration, document the interconnection of the PROFIsafe1 and DIAG_1 outputs, along with the value of the LADDR input
2. For redundant F-I/O, also document the interconnection of the PROFIsafe2 and DIAG_2 outputs before performing the migration, along with the value of the LADDR_R input
3. Perform the migration to *S7 F Systems Lib V1_3 SP2*.
4. Interconnect the documented interconnections at the PROFIsafe1 and DIAG_1 outputs to the new module driver F_PS_12, whose value at the LADDR input matches the documented LADDR
5. For redundant F-I/O, interconnect the documented interconnections at the PROFIsafe2 and DIAG_2 outputs to the new F-Module driver F_PS_12, whose value at the LADDR input matches the documented LADDR_R

| <i>Failsafe Blocks (V1_2)</i> | <i>S7 F Systems Lib V1_3 SP2</i> |
|---|--|
| Interconnection at the original F-Module driver: | Interconnection at the F-Module driver F_PS_12: |
| PROFIsafe1 | PROFIsafe |
| DIAG_1 | DIAG |
| LADDR | LADDR |

| <i>Failsafe Blocks (V1_2)</i> | <i>S7 F Systems Lib V1_3 SP2</i> |
|---|--|
| Redundant interconnection at the original F-Module driver: | Interconnection at the 1st F-module driver F_PS_12: |
| PROFIsafe1 | PROFIsafe |
| DIAG_1 | DIAG |
| LADDR | LADDR |
| | Interconnection at the 2nd F-module driver F_PS_12: |
| PROFIsafe2 | PROFIsafe |
| DIAG_2 | DIAG |
| LADDR_R | LADDR |

Additional measures for redundant fail-safe digital input modules SM 326; DI 8 X NAMUR and SM 326; DI 24 X DC 24 V

Starting with *S7 F Systems Lib V1_3 SP1*, discrepancy error information is output at the DISCF and DISCF_R outputs of the F-Channel driver F_CH_DI.

If you are using logic that evaluates this information, modify it accordingly.

4.3.4 User scenario 4

Objective

Upgrade of your S7 program with *S7 F Systems Lib V1_3* to *S7 F Systems Lib V1_3 SP2*.

Introduction

This user scenario helps you when migrating your safety program by upgrading blocks of the *S7 F Systems Lib V1_3* to blocks of the *S7 F Systems Lib V1_3 SP2* F-Library, enabling you to use the new functions of the *S7 F Systems Lib V1_3 SP2* F-Library.

When you migrate from *S7 F Systems Lib V1_3* to *S7 F Systems Lib V1_3 SP2*, the F-FBs in your safety program are overwritten by F-blocks with different block signatures. This means that the collective signature will change.

Requirement

If F-block types are used in your project, you must re-create these with *S7 F Systems Lib V1_3 SP2* beforehand. To do this, follow the procedure outlined in section "Updating custom F-block types (Page 43)".

Consequences

For possible consequences, refer to the section "Acceptance test following system upgrade (Page 211)".

Procedure

1. Install *S7 F Systems V6.2* with *S7 F Systems Lib V1_3 SP2*.
2. Prior to the initial compilation, save the current version of your safety program as a reference ("Save reference" in the "Safety program" dialog) so that it will be available for future comparisons.
3. In the Safety Program dialog, select the *S7 F Systems Lib V1_3* F-Library.
To do so, click the "Library Version" button in the "Edit safety program" dialog.
4. In the S7 program, update the existing F-block types. See section "Updating custom F-block types (Page 43)" for more on this.

5. Update all block types in the *CFC Editor* by selecting **Options > Block Types** and clicking "New Version".
6. In the *CFC Editor* under **Options > Block Types**, click "Clean Up".
7. Compile your hardware configuration.
8. Compile your S7 program.

4.3.5 Updating custom F-block types

If F-block types are used in your project, you must re-create these with *S7 F Systems Lib V1_3 SP2*. To do so, you require the project in which the F-block type was created with the menu command **Chart > Compile > Chart as block type** in the *CFC Editor* (source project).

Proceed as follows:

1. Install *S7 F Systems V6.2* with *S7 F Systems Lib V1.3 SP2*.
2. In the Safety Program dialog, select the *S7 F Systems Lib V1_3* F-Library.
3. Update all block types in the *CFC Editor* by selecting **Options > Block Types** and clicking "New Version".
4. In the *CFC Editor* under **Options > Block Types**, click "Clean Up".
5. Open the CFC chart to be compiled and compile it in the *CFC Editor* using the menu command **Chart > Compile > Chart as block type**.
6. You can now copy the compiled F-block type to your S7 programs in which you want to use it.

See also

Creating F-Block types (Page 95)

4.3.6 Updating a multiproject master data library

Introduction

The following describes how you apply the F-blocks from *S7 F Systems Lib V1.3 SP2* to the master data library of your multiproject.

Requirement

The user projects are already updated.

Note

You update the user projects in your multiproject as described in section "Migration to S7 F Systems V6.2 (Page 32)".

If you are using F-block types that you have created in your master data library, you must update these F-block types as described in section "Updating custom F-block types (Page 43)".

All attributes of the F blocks must be applied. Do not perform an update of the old F-block attributes.

Procedure

Proceed as follows to continue using the master data library with fail-safe blocks as usual in the multiproject:

1. Open the block folder in the master data library of your multiproject and select the "Details" view option.
2. Delete all blocks with the author "F_SAFE11" or "F_SAFE12".
Important: Select the "Also delete symbolic block names" option.
3. In *SIMATIC Manager*, select **File > Open** and switch to the "Libraries" tab.
4. Select the "*S7 F Systems Lib V1_3*" library and confirm with "OK".
Result: The library opens.
5. Select the "F-User Blocks" library component to be copied. Select the **Edit > Copy** menu command.
6. Select the folder in the master data library (destination) in which the copied library component is to be placed.
7. Select the menu command **Edit > Paste**. The copied library component is placed into the master data library.
8. Repeat Steps 3 to 5 for the "F-Control Blocks" library component.
9. Repeat Steps 3 to 5 for the block folder containing the F-block types that you created.
10. In *SIMATIC Manager*, select **Options > Charts > Update Block Types** for the master data library. This will update all blocks in your sample solutions and process tag types in the master data library.

Configuration

5.1 Configuration overview

Introduction

The following section lists the main points in which the configuration of a fail-safe system differs from that of an S7 standard system.

Fail-safe components that you must configure

You must configure the following hardware components for *S7 F Systems V6.2*:

1. F-CPU, such as CPU 417-5H
2. F-I/O, such as:
 - ET 200S fail-safe I/O modules
 - ET 200SP fail-safe I/O modules
 - S7-300 fail-safe signal modules in ET 200M (distributed configuration)
 - ET 200eco fail-safe I/O modules
 - ET 200pro fail-safe I/O modules
 - ET 200iSP fail-safe I/O modules
 - Fail-safe DP standard slaves
 - Fail-safe IO standard devices
 - Fail-safe PA field devices

Note

F-I/O for PCS 7

For information on the F-I/O released for PCS 7, please refer to the "Process Control System PCS 7 Released Modules" manual for the respective PCS 7 version.

5.2 Particularities for configuring an F-System

Configuring same as in standard system

You configure an S7 F/FH Systems fail-safe system the same as a standard S7 system. That is, you configure and assign parameters for the hardware in *HW Config* as a centralized configuration (F-CPU) and as a distributed configuration (F-CPU, F-SMs in ET 200M, F-modules in ET 200S, ET 200SP, ET 200pro, ET 200iSP and ET 200eco, fail-safe DP standard slaves/IO standard devices, fail-safe PA field devices).

For a detailed description of the configuration variants, refer to the "Safety Engineering in SIMATIC S7 (<http://support.automation.siemens.com/WW/view/en/12490443>)" system manual.

Special F-relevant tabs

There are a few special tabs for the fail-safe functionality in the object properties of the fail-safe I/O. These tabs are described in the following sections.

Assigning symbols for fail-safe inputs/outputs of the fail-safe I/O

For convenience when programming S7 F/FH Systems, it is particularly important that you assign symbols for the fail-safe inputs and outputs of the F-I/O in *HW Config*.

Saving and compiling the hardware configuration

You must save and compile the hardware configuration of S7 F/FH systems in *HW Config*. This is required for subsequent programming of the safety program.

Changing safety-related parameters

Note

If you change a safety-related parameter for an F-I/O or an F-CPU, you must recompile the S7 program.

The same applies to changes in S7 connections for safety-related communication via S7 connections.

Rules for F-systems

In addition to the rules that are generally applicable for the arrangement of modules in an S7-400, you must also comply with the following conditions for an F-system:

- Prior to downloading the safety program, you must download the hardware configuration to the F-CPU.
- If you have changed the configuration of an F-I/O or the F-CPU (cycle times of the cyclic-interrupt OB), you must recompile the S7 program and download it to the F-CPU.

5.3 Configuring the F-CPU

Rules for configuring an F-CPU

WARNING

An F-CPU containing a safety program must have a password.

You must ensure that the following conditions are met:

- The "CPU contains safety program" option must be selected.
- A password must *always* be assigned.

You make these settings in the object properties of the F-CPU in *HW Config*.

WARNING

Configuring a protection level

In safety mode, access authorization by means of the F-CPU password must not be active when changes are made to the standard user program, because the safety program can then also be changed. To rule out this possibility, you must configure protection level "1". If only one person is authorized to change the standard user program and the safety program, protection level "2" or "3" should be configured to ensure that other persons have only limited access or no access to the standard user program and safety program.

Procedure for configuring the protection level

To configure the protection level 1, follow the steps below:

1. Select the desired F-CPU in *HW Config*, e.g. CPU 417-5H, and then the menu command **Edit > Object properties**.
2. Open the "Protection" tab.
3. Set the protection level to "1: Access protection for F-CPU or keyswitch setting" and "Can be bypassed with password".

Enter a password for the F-CPU in the fields provided for that purpose and activate the "CPU contains safety program" option.

You can find information on the password for the F-CPU in section "Overview of access protection (Page 69)". In particular, observe the warning in section "Setting up access rights for the F-CPU (Page 71)":

Furthermore, it is recommended that the increased password security option be used. The increased password security is only relevant for the engineering system. When this option is activated, the entered password is stored encrypted in the data management. That increases the security of the password. Setting this option does not affect the response to a password operation.

Important parameters for the F-CPU in S7 FH Systems

To prevent the time monitoring from responding at a master-standby switchover (e.g. H-CiR), you must configure the OB 3x(s) designated for safety programs with a priority > 15 in the "Cyclic interrupts" tab of the F-CPU. You should not place standard blocks in these OBs.

The cyclic interrupt OB of the safety program must be configured as "Cyclic interrupt OB with special handling". Only then will this cyclic interrupt be called just before the start of the disabling time for priority classes > 15 when the standby CPU is updated. For this purpose, you enter the number of the highest priority cyclic interrupt OB to which F-blocks of the safety program are assigned in the *CFC Editor* in the "Cyclic interrupt OB with special handling" field on the "H Parameters" tab of the CPU properties.

- Ensure that the correction factor is set to 0 ms in the "Clock" group on the "Diagnostics/Clock" tab.

Note

For S7 FH Systems, only settings up to 12 hours are allowed.

For S7 FH Systems, you are not permitted to modify safety-related self-tests via SFC 90 "H_CTRL". Otherwise, the safety program goes to F-STOP after 24 hours at the latest. Test components are not permitted to be switched on or off (submode 0 to 5 of mode 20, 21 and 22).

For the same reason, you must not disable the updating via SFC 90 "H_CTRL" too long.

Failure to observe this will trigger an F-STOP. The following diagnostics event is then entered in the diagnostics buffer of the F-CPU:

- "Safety program: Error detected" (event ID 16#75E1)
-

Changing the OB3x cycle time

After a change of the OB 3x cycle times, you must recompile the S7 program.

5.4 Configuring the F-I/O

5.4.1 Overview of configuring the F-I/O

Overview

The configuring of the F-I/O differs in the following options:

- The `F_destination_address` is set on the fail-safe module via a DIP switch. This option applies to F-modules such as ET 200S, ET 200eco, ET 200pro, and ET 200iSPs and the S7-300 F-SMs.

You can find additional information on this in section "Configuring the fail-safe modules with assignment of `F_destination_address` via DIP switch (Page 49)".

- For F-modules such as ET 200SP, the PROFIsafe address (F-source address and F-destination address) is assigned directly from the Engineering System in STEP 7.

These fail-safe modules do not have a DIP switch for setting the unique F-destination address for each module.

You can find additional information on this in section "Configuring fail-safe modules with assignment of `F_destination_address` in the Engineering System (Page 52)".

5.4.2 Configuring the fail-safe modules with assignment of `F_destination_address` via DIP switch

Configuring same as in standard system

F-modules such as ET 200S, ET 200eco, ET 200pro, ET 200iSP and the S7-300 F-SMs are always configured in the same way:

Once the F-I/O have been inserted into the station window of *HW Config*, you can access the configuration dialog by selecting **Edit > Object Properties** or by double-clicking the F-I/O.

When changes are made to fail-safe I/O in *HW Config*, you will be prompted to enter the password for the safety program.

The values in the shaded fields are automatically assigned by *S7 F Systems* in the F-relevant tab. You can change the values in the non-shaded fields.

Additional Information

For information on which ET 200S, ET 200eco, ET 200pro, and ET 200iSP F-modules and which S7-300 F-SMs you can use, refer to the "S7 Distributed Safety, configuring and programming (<http://support.automation.siemens.com/WW/view/en/22099875>)" system manual.

For a description of the parameters, refer to the *context-sensitive online help* for the tab and the relevant *F-I/O manual*.

For information on what you must consider when configuring the F-monitoring time for fail-safe I/O, refer to the "Safety Engineering in SIMATIC S7 (<http://support.automation.siemens.com/WW/view/en/12490443>)" system manual.

Assigning symbols for fail-safe inputs/outputs of the fail-safe I/O

For convenience when programming S7 F/FH Systems, it is important that you assign symbols for the fail-safe inputs and outputs of the F-I/O in *HW Config*.

Note that for certain fail-safe I/O (such as S7-300 F-SMs and ET 200S F-modules), a 1oo2 evaluation can be set for the sensor. In this case, only one of the two combined channels is available.

We recommend that you identify the unavailable channel as reserved in the symbol table. To find out which of the channels combined by the 1oo2 sensor evaluation you can access in the safety program, refer to the relevant manuals for the F-I/O.

Operating mode

For S7-300 fail-safe signal modules, you can distinguish on the basis of the "Operating mode" parameter whether the modules are being used in standard mode (used as standard S7-300 signal modules SM 326; DO 8 × DC 24V/2A, SM 326 F DO 10xDC24V/2A PP (6ES7326-2BF10-0AB0) and SM 336 F AI6x0/4..20mA HART (6ES7336-4GE00-0AB0)) or in safety mode.

ET 200S, ET 200pro, ET 200iSP and ET 200eco fail-safe modules can only be used in safety mode.

Group diagnostics for fail-safe S7-300 signal modules

The "Group diagnostics" parameter is used to activate and deactivate the transmission of channel-specific diagnostic alarms of F-SMs (such as wire break and short-circuit) to the F-CPU. For availability reasons, you should shut down the group diagnostics on unused input or output channels of the following F-SMs:

- SM 326; DI 8 x NAMUR
- SM 326; DO 10 x DC 24V/2A
- SM 336; AI 6 x 13Bit



WARNING

"Group diagnostics" for fail-safe F-SMs in safety mode

"Group diagnostics" must be activated on all connected channels of fail-safe F-SMs in safety mode.

Check to verify that you have shutdown group diagnostics only for unused input and output channels.

You can optionally enable diagnostic interrupts.

The following applies to modules:

- SM 326; DI 24 x DC 24V (as of article no. 6ES7326-1BK01-0AB0)
- SM 326, F-DO10 x DC24V/2A PP (6ES7326-2BF10-0AB0)
- SM 326; DO 8 x DC 24V/2A PM (as of article no. 6ES7326-2BF40-0AB0)
- The following applies to SM 336, F-AI 6 x 0/4 ... 20mA HART (6ES7336-4GE00-0AB0):

By disabling a channel in *HW Config* you also disable its group diagnostics function.

PROFIsafe addresses

The PROFIsafe addresses (assigned `F_source_address` and `F_destination_address` parameters) uniquely identify the source and destination.

F_destination_address

The `F_destination_address` uniquely identifies the PROFIsafe destination (of the F-I/O). Therefore, the `F_destination_address` must be unique network-wide and station-wide (see section "Rules for address assignment").

To prevent incorrect parameter assignment, a *station-wide unique* `F_destination_address` is automatically assigned when the F-I/O are placed in *HW Config*.

In S7 F/FH Systems, you must ensure that the `F_destination_address` is *unique network-wide* when multiple stations are present in a network by manually changing the `F_destination_addresses`.

If you change the `F_destination_address`, the uniqueness of the `F_destination_address` within the station is checked automatically. You yourself must make sure that the `F_destination_address` is unique network-wide.

You must set the `F_destination_address` on the F-I/O via the DIP switch before installing the F-I/O.

Note

For the following S7-300 F-SMs, the `F_destination_address` is the same as the start address of the F-SM/8:


- SM 326; DI 24 x DC 24V (article no. 6ES7326-1BK00-0AB0),
- SM 326; DI 8 x NAMUR (article no. 6ES7326-1RF00-0AB0)
- SM 326 DO 10 x DC 24V/2A (article no. 6ES7326-2BF01-0AB0)
- SM 336; AI 6 x 13-bit (article no. 6ES7336-1HE00-0AB0)

Assign low start addresses for these F-SMs if you are also using other F-I/O.

F_source_address

The `F_source_address` is automatically assigned in *S7 F Systems* and is preset with the value "1".

Rules for address assignment

| |
|---|
|  WARNING |
| <p>Address assignment in subnets only and in mixed configurations</p> <p>The following applies to PROFIBUS DP subnets only:</p> <p>The PROFIsafe destination address and, thus, the switch setting on the address switch of the F-I/O must be unique network-wide* and station-wide** (system-wide).</p> <p>For S7-300 F-SMs and ET 200S, ET 200eco, ET 200iSP and ET 200pro F-modules, you can assign a maximum of 1022 different PROFIsafe destination addresses.</p> <p>The following applies to PROFINET IO subnets only and to mixed configurations of PROFIBUS DP and PROFINET IO:</p> <p>The PROFIsafe destination address and, thus, the address switch setting on the F-I/O must be unique only*** within the PROFINET IO subnet, including all lower-level PROFIBUS DP subnets, and station-wide** (system-wide).</p> <p>For S7-300 F-SMs and ET 200S, ET 200eco, ET 200iSP and ET 200pro F-modules, you can assign a maximum of 1022 different PROFIsafe destination addresses.</p> <p>A PROFINET IO subnet is characterized by the fact that the IP addresses of all networked nodes have the same subnet address, i.e. the IP addresses match in the positions that have the value "1" in the subnet mask.</p> <p>Example:</p> <p>IP address: 140.80.0.2.</p> <p>Subnet mask: 255.255.0.0 = 11111111.11111111.00000000.00000000</p> <p>Meaning: Bytes 1 and 2 of the IP address define the subnet; subnet address: 140.80.</p> <p>* A network consists of one or more subnets. "Network-wide" means across subnet boundaries.</p> <p>** "Station-wide" means for one station in <i>HW Config</i> (e.g. an S7-400H station).</p> <p>*** Across Ethernet subnets, excluding cyclic PROFINET IO communication (RT communication)</p> |

5.4.3 Configuring fail-safe modules with assignment of F_destination_address in the Engineering System

Note

Note that the rules for address assignment of the PROFIsafe destination addresses also apply to fail-safe modules such as ET 200SP. Please observe the warning in section "Configuring the fail-safe modules with assignment of F_destination_address via DIP switch (Page 49)":

Introduction

Fail-safe modules, such as in the ET 200SP, have no DIP switch for assigning the unique F-destination address for each module.

Instead, you assign the PROFIsafe address (F-source address and F-destination address) directly from the Engineering System in STEP 7. Before you use a fail-safe module, you must assign it the associated F-destination address together with the F-source address.

For fail-safe modules, e.g. of ET 200SP, a reassignment is necessary in the following cases:

- Subsequent insertion of an F-module during first commissioning
- Deliberate change to the F-destination address
- Replacement of the coding element
- Commissioning of a series machine

A reassignment is not necessary in the following cases:

- Power OFF/ON
- Replacement of an F-module (repair case) without PG/PC
- Replacement of the BaseUnit
- Change to the configuration if a new BaseUnit is inserted before an F-module
- Repair/replacement of the interface module

Basic procedure

1. Configure the F-destination address in HW Config.
2. Identify the F-modules in the distributed I/O, e.g. ET 200SP, to which you want to assign the configured F-destination addresses (together with the F-source address).

You can find additional information on this in section "Identifying fail-safe modules (address assignment in the ES) (Page 53)".

3. Assign the F-destination address (together with the F-source address) to the F-modules.

You can find additional information on this in section "Assigning F-destination address and F-source address (address assignment in the ES) (Page 55)".

5.4.3.1 Identifying fail-safe modules (address assignment in the ES)

Requirements

The following requirements must be met:

- The F-module, e.g. ET 200SP, is configured.
- The configuration was downloaded to the F-CPU.
- The F-module can be reached online.

 **WARNING**

Identification and confirmation of the F-I/O

By pressing the "Identification" button, you confirm the fail-safe correctness of the PROFIsafe addresses for the F-I/O.

Therefore, proceed cautiously when confirming the F-I/O by LED flashing or by the serial number of the interface module.

Procedure

To identify the fail-safe modules, such as ET 200SP, follow the steps below:

1. Establish an online connection to the F-CPU on which this F-module will be operated.
2. In HW Config, select this F-module together with the F-modules to which you want to assign the F-destination address (together with the F-source address).
3. Select "Name F-destination address" from the shortcut menu.
4. Select the method for identifying the F-modules under "Assign F-destination address by".

- "Identify by LED flashing"

This is the default setting. During the identification, the DIAG and STATUS LEDs of the F-modules to be identified flash.

- "Identify by serial number"

If you do not have direct sight of the F-modules, you can identify the F-modules using the serial number of the interface module.

Note

Unlike the serial number printed on the interface module, the displayed serial number may be supplemented to include the year date. The serial numbers are identical despite that.

5. In the "Assign" column, select all F-modules to which you want to assign the F-destination address. If you select the "Assign F-destination address for all accessible ET 200SP", all F-modules of the station will be selected.
6. Click the "Identification" button. Observe whether the DIAG LED and status LEDs of the F-modules whose F-destination address you want to assign are flashing green. If you use serial number for identification, compare the displayed serial number with the serial number of the interface module.

See also

Configuring fail-safe modules with assignment of F_destination_address in the Engineering System (Page 52)

Assigning F-destination address and F-source address (address assignment in the ES) (Page 55)

5.4.3.2 Assigning F-destination address and F-source address (address assignment in the ES)**Requirement**

The F-modules such as ET 200SP have been successfully identified as described in section "Identifying fail-safe modules (address assignment in the ES) (Page 53)".

Procedure

To assign the F-destination address and F-source address, follow these steps:

1. Assign the F-destination address and the F-source address to the fail-safe modules with the "Assign F-destination address" button. You must enter the password of the F-CPU if necessary.
2. To assign the F-destination address together with the F-source address, you must confirm the "Confirm assignment" dialog within 60 seconds.

See also

Configuring fail-safe modules with assignment of F_destination_address in the Engineering System (Page 52)

5.4.3.3 Changing the F-destination address or F-source address (address assignment in the ES)**Procedure**

1. Change the F-destination address or F-source address in the hardware configuration.
2. Compile the hardware configuration.
3. Download the hardware configuration to the F-CPU.
4. Select "Assign F-destination address" in the shortcut menu.
5. Repeat the steps described in sections "Identifying fail-safe modules (address assignment in the ES) (Page 53)" and "Assigning F-destination address and F-source address (address assignment in the ES) (Page 55)".
6. Compile the user program and download it to the F-CPU.

5.5 Configuring fail-safe DP standard slaves/IO standard devices

Requirement

In order to use fail-safe DP standard slaves/IO standard devices, these standard devices must be on PROFIBUS DP or PROFINET IO and support the PROFIsafe bus profile.

Configuring with GSD/GSDML file

As is the case in a standard system, the fail-safe standard slaves are configured based on the device specification in the so-called GSD file (generic station description).

- In the GSD file for DP standard slaves.
- In der GSDML file for IO standard devices.

For these fail-safe standard devices, portions of the specification are protected by a cyclic redundancy check (CRC).

The GSD/GSDML files are supplied by the device manufacturers. The supplied GSD/GSDML file must satisfy the PROFIsafe Specification in order for fail-safe standard devices to be operated with *S7 F Systems*.

- For fail-safe DP standard slaves PROFIsafe Specification V1.0 or higher
- For fail-safe IO standard devices: PROFIsafe Specification V2.0 to V2.4

Ask for confirmation of this from the device manufacturer.

Import the GSD/GSDML files into your project (see *STEP 7* online help). Once the fail-safe standard devices are imported, they can be selected in the hardware catalog of *HW Config*.

Protection of the data structure of the device in GSD/GSDML files

Starting with PROFIsafe Specification V2.0, the data structure of the device described in the GSD or GSDML file must be protected with a CRC stored in this file ("setpoint" for F_IO_StructureDescCRC).

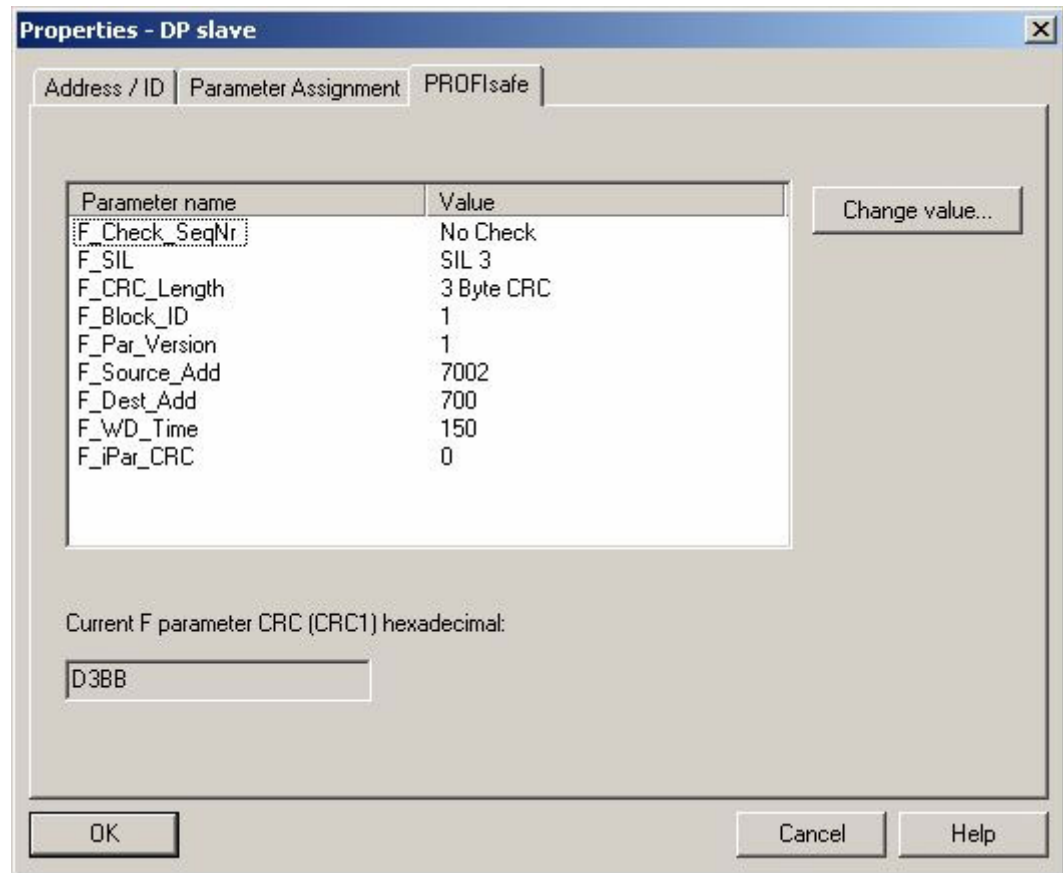
Procedure for configuring with GSD/GSDML file

Import the GSD/GSDML file into your project (see *STEP 7* online help).

1. Select the corresponding fail-safe DP standard slave, e.g. a DP standard slave, in the hardware catalog of *HW Config* and insert it into your DP master system/PROFINET IO system.
2. Select the fail-safe DP or IO master.
3. Open the object properties dialog using the **Edit > Object Properties** menu command or by double-clicking the slot of the fail-safe component.

Channel-level passivation is not supported for fail-safe DP standard slaves.

The figure below shows the properties of a DP standard slave as an example.



"PROFIsafe" tab

The parameter texts specified in the GSD/GSDML file are contained on the "PROFIsafe" tab under "Parameter name". The associated current value is shown under "Value". You can modify this value using the "Change value" button.

The parameters are explained below.

Parameter "F_Check_SeqNr"

This parameter defines whether the sequence number is to be incorporated in the consistency check (CRC calculation) of the F-User data frame.

The "F_Check_SeqNr" parameter must be set to "No check" in the PROFIsafe V1 MODE. Only fail-safe DP standard slaves/PA field devices that behave accordingly are supported.

"F_CHECK_SeqNr" is irrelevant in PROFIsafe V2 MODE.

Parameter "F_SIL"

Sicherheitsklasse des fehlersicheren DP-Normslaves/IO-Normdevice/PA-Feldgerätes. The parameter is device-dependent. Possible settings for the "F_SIL" parameter are "SIL 1" to "SIL 3", depending on the GSD/GSDML file.

Parameter "F_CRC_Length"

A cyclic redundancy check with a length of 2 bytes, 3 bytes or 4 bytes is required, depending on the length of the F-user data (process data) and the PROFIsafe mode. This parameter provides information to the F-CPU on the size of the CRC2 key in the safety message frame.

In PROFIsafe V1 MODE:

For a user data length less than or equal to 12 bytes, select 2-byte CRC as the setting for the "F_CRC_Length" parameter; for a user data length ranging from 13 bytes to 122 bytes, select 4-byte CRC.

S7 F Systems only supports a user data length up to and including 12 bytes and "2-byte CRC". The fail-safe DP standard slave/PA field device must behave accordingly.

In PROFIsafe V2 MODE:

For a user data length less than or equal to 12 bytes, select 3-byte CRC as the setting for the "F_CRC_Length" parameter; for a user data length ranging from 13 bytes to 123 bytes, select 4-byte CRC.

S7 F Systems only supports a user data length up to an including 12 bytes and "3-byte CRC". The fail-safe DP standard slave/IO standard device/PA field device must behave accordingly.

Parameter "F_Block_ID"

The F_Block_ID parameter has the value 1 if the F_iPar_CRC parameter exists, otherwise it has the value 0.

The value 1 of the F_Block_ID parameter indicates that the data record for the value of F_iPar_CRC has been extended by 4 bytes. You must not change parameters.

Parameter "F_Par_Version"

This parameter identifies the PROFIsafe operating mode. You can identify the operating modes supported by the device from the value range offered.

For fail-safe IO standard devices, this parameter is set to "1" (PROFIsafe V2 MODE) and cannot be changed.

For fail-safe DP standard slaves/PA field devices, you can set this parameter to the following:

- Set "F_Par_Version" to "1" (PROFIsafe V2 MODE) for a homogenous PROFIBUS DP network, if the device and the F-CPU support this. Otherwise, set it to "0" (PROFIsafe V1 MODE).
- For a network that consists of PROFIBUS DP and PROFINET IO subnets, "F_Par_Version" must be set to "1" (PROFIsafe V2 MODE).

Note

The following F-CPU's support V2 MODE:

- CPU 412-3H (article no. 6ES7412-3HJ14-0AB0) and higher; firmware V4.5 and higher
- CPU 414-4H (article no. 6ES7414-4HM14-0AB0) and higher; firmware V4.5 and higher
- CPU 416-5H (article no. 6ES7 416-5HS06-0AB0) and higher; firmware V6.0 and higher
- CPU 417-4H (article no. 6ES7417-4HT14-0AB0) and higher; firmware V4.5 and higher
- CPU 410-5H (article no. 6ES7 410-5HX08-0AB0) and higher

If you set "F_Par_Version" to "1" for F-CPU's that do not support PROFIsafe V2 MODE, this will result in a communication error during safety-related communication with the device. One of the following diagnostics events is then entered in the diagnostics buffer of the F-CPU:

- "F-I/O passivated": Check value error (CRC)/Sequence number error ...
 - "F-I/O passivated": F-Monitoring time exceeded at the safety message frame detected in the F-CPU ...
-



WARNING

Devices and "F_Par_Version" parameter for a mixed configuration

For a network that consists of PROFIBUS DP and PROFINET IO subnets, "F_Par_Version" must be set to "1" (PROFIsafe V2 MODE).

Devices that do not support PROFIsafe V2 MODE must not be used on a PROFINET IO network only or with mixed configurations of PROFIBUS DP and PROFINET IO.

Parameters "F_Source_Add" and "F_Dest_Add"

The PROFIsafe addresses ("F_Source_Add" and "F_Dest_Add" parameters) uniquely identify the source and destination.

The "F_Source_Add" and "F_Dest_Add" parameters for fail-safe DP standard slaves/IO standard devices/PA field devices correspond to the "F_source_address" and "F_destination_address" parameters of other F-I/O. Exception: The value range is specified by the GSD/GSDML file and is not limited to 1 to 1022 for the PROFIsafe destination address.

Therefore, the information about PROFIsafe address assignment provided in section "Configuring the fail-safe modules with assignment of F_destination_address via DIP switch (Page 49)" is generally applicable to fail-safe DP standard slaves/IO standard devices.

Parameter "F_WD_Time"

This parameter defines the F-monitoring time in the fail-safe DP standard slave/IO standard device/PA field device.

A valid current safety message frame must come from the F-CPU within the monitoring time period. This ensures that failures and faults are detected and appropriate reactions are triggered to maintain the fail-safe system in a safe state or bring it to a safe state.

You should set the monitoring time high enough for the message frame delays to be tolerated by the communication, but low enough for the fault reaction function to react quickly in case of a fault (interruption of the communication connection, for example). (See "Safety Engineering in SIMATIC S7" system manual).

The "F_WD_Time" parameter can be set in 1 ms increments. The value range of the "F_WD_Time" parameter is specified by the GSD/GSDML file.

For more information about the F-monitoring time, refer to section "Run times, F-Monitoring times, and response times (Page 460)".

Parameter "F_iPar_CRC"

CRC via individual device parameters (i-parameter).

The individual device parameters (i-parameters) of a fail-safe DP standard slave/IO standard device/PA field device are configured using the device manufacturer's own parameterization tool.

Enter the CRC calculated by the parameterization tool of the device manufacturer for protection of the i-parameters. *S7 F Systems* takes the value into account when calculating the F-Parameter CRC (CRC1) .

See also

Safety engineering in SIMATIC S7
(<http://support.automation.siemens.com/WW/view/en/12490443>)

5.6 Configuring fail-safe PA field devices

Fail-safe PA field devices are configured in the same way as fail-safe DP standard slaves.

When configuring PA field devices, follow the procedure described in the chapter entitled "Configuring fail-safe DP standard slaves/IO standard devices (Page 56)".

5.7 Configuring redundant F-I/O

Introduction

To increase availability of your automation system and, thus, to prevent process failures due to faults in the fail-safe system, you can optionally equip S7 F/FH Systems fail-safe systems as fault-tolerant systems (S7 FH Systems). This increased availability can be achieved by component redundancy (F-CPU, communication connection and F-I/O).

For S7 F Systems, availability can be increased without fault-tolerant configuration. You can use S7-300 fail-safe signal modules (F-SMs) redundantly in one ET 200M or in several ET 200Ms.

Note

In the case of redundantly configured F-SMs, you must ensure the following:

- Both F-SMs must be of the same type, product version and firmware
 - "Safety mode" must be activated as the operating mode for both F-SMs on the "Parameters" tab of the object properties dialog.
-

Procedure

To configure two S7-300 fail-safe signal modules redundantly, for example, follow these steps:

1. In *HW Config*, configure the two F-SMs in the ET 200M(s).
2. Configure the first F-SM:
Activate the "Safety mode" operating mode on the "Parameters" tab
3. Configure the second F-SM:
Activate the "Safety mode" operating mode on the "Parameters" tab
4. For the second F-SM, set the "2 modules" operating mode on the "Redundancy" tab.
5. Select the first F-SM for the F-SM in the "Find redundant module" dialog.
6. Set additional parameters, if necessary. The settings are applied automatically for the first F-SM. As soon as two F-SMs are redundant, changes in the parameter assignment for one of the F-SMs are applied automatically for the other F-SM.
7. For redundant fail-safe digital input modules, the F-channel driver F_CH_DI can perform a discrepancy analysis for increased availability. You must set the "Discrepancy time" parameter for this. If you set discrepancy time "0", the discrepancy analysis is deactivated. You can find additional information in the online help for the "Redundancy" tab.
8. For redundant fail-safe analog input modules, the F-channel driver F_CH_AI can perform a discrepancy analysis for increased availability. You must activate the "DISC_ON" parameter for this. This setting must be made in the CFC. You can find additional information in the online help for the "Redundancy" tab.

See also

F_CH_DI: F-channel driver for digital inputs of F-I/O (except fail-safe DP standard slaves and fail-safe IO standard devices) (Page 328)

F_CH_AI: F-channel driver for analog inputs of F-I/O (except fail-safe DP standard slaves and fail-safe IO standard devices) (Page 336)

5.8 System modifications during operation

Introduction

Certain process control systems must not be shut down during operation. This is due to the complex nature of automation systems or the high cost of a restart, for example. At certain times, however, these systems do require changes or expansions. This is possible with Configuration in RUN mode (CiR for short). With CiR, the program sequence is stopped for up to 2500 ms. The process outputs retain their current value during this period. This has no effect on the actual process, especially in process control systems.

System change during operation by means of CiR is based on provisions in the master system of the initial configuration for a subsequent hardware expansion of your automation system. You define suitable CiR elements that you can later replace with real elements on a step-by-step basis in RUN mode. You can download a configuration modified in this way to the F-CPU during process operation.

Before performing the procedures described below, read the CiR instructions in manual "Modifying the System during Operation via CiR" (<https://support.industry.siemens.com/cs/en/en/view/45531308>).

Calculating F-Monitoring times

When calculating the minimum F-Monitoring times, take the CiR synchronization time into account. Refer also to the section entitled "Run times, F-Monitoring times, and response times (Page 460)".

Reducing F-Monitoring times

If the calculated values for the process are not acceptable, you can recalculate the F-Monitoring time by reducing the CiR synchronization time. You have the following options for doing so:

- Decrease the number of input and output bytes of the master system.
- Decrease the number of guaranteed slaves of the master systems that you intend to change.
- Decrease the number of master systems that you intend to change during a CiR.

Extending the maximum cycle time using CiR

If CiR is used, the maximum cycle time is extended by the *lesser* of the following two values:

- CiR synchronization time of F-CPU

The CiR synchronization time of the F-CPU is the sum of the CiR synchronization times for all DP master systems that are to be changed simultaneously. The CiR synchronization time of a DP master system is displayed in *HW Config* in the Properties dialog for the relevant CiR object.

- Upper limit of CiR synchronization time

The default value for this upper limit is 1 second. You can increase or decrease this value according to your requirements by calling SFC 104 "CiR".

For instructions on determining the maximum cycle time, refer to the manual for the F-CPU you are using.

Limiting CiR synchronization time:

The F-CPU compares the actually calculated CiR synchronization time with the current upper limit for the CiR synchronization time. If the calculated value is less than the current upper limit, CiR is enabled. The default value for the upper limit of the CiR synchronization time in the F-CPU is 1 second. SFC 104 enables you to change this value. You can raise or lower the upper limit within a range of 200 ms to 2500 ms. For a detailed description of SFC 104, refer to manual " System Software for S7-300/400 System and Standard Functions (<http://support.automation.siemens.com/WW/view/en/1214574>) ".

5.8.1 Configuring F-I/O with CiR

Introduction

CiR allows you to add new fail-safe I/O to your system or delete existing fail-safe I/O from your system. The following two sections explain the procedure.

Adding fail-safe I/O with CiR

Add fail-safe I/O to your system as follows:

1. Configure the new fail-safe I/O in *HW Config*. Follow the procedure as described in the "Modifying the System during Operation via CIR (<https://support.industry.siemens.com/cs/en/en/view/45531308>)" manual. Handle the fail-safe I/O the same as standard I/O.
2. Extend your S7 program and compile it with the "Changes" scope and activated "Generate module drivers" option.
3. Download your safety program.
4. When safety mode is activated, you are prompted whether you want to disable safety mode.

Confirm this prompt. Safety mode is deactivated and the download operation is carried out.

Note

A user acknowledgment at the ACK_REI input is required to activate the fail-safe I/O.

5. After completion of the download operation, you are prompted whether you want to activate safety mode. Confirm this prompt.

Safety mode is activated.

Note

Parameter reassignment of fail-safe I/O is not supported. Additional information can be found in the "Fault-Tolerant Systems S7-400H (<http://support.automation.siemens.com/WW/view/en/82478488>)" system manual.

Deleting fail-safe I/O with CiR

Delete fail-safe I/O from your system as follows:

1. Delete the fail-safe I/O in *HW Config*. Follow the procedure as described in the "Modifying the System during Operation via CiR (<https://support.industry.siemens.com/cs/en/en/view/45531308>)" manual. Handle the fail-safe I/O the same as standard I/O.
2. Change your S7 program and compile it with the "Changes" scope and activated "Generate module drivers" option.
3. Download your safety program.
4. When safety mode is activated, you are prompted whether you want to disable safety mode.

Confirm this prompt. Safety mode is deactivated and the download operation is carried out.

5. Download your configuration using CiR.
6. After completion of the download operation, you are prompted whether you want to activate safety mode.

Confirm this prompt. Safety mode is activated.

Note

You can delete an existing fail-safe I/O using CiR only if the fail-safe I/O is assigned to a CiR object in the relevant master system.

See also

Deactivating safety mode (Page 195)

Activating safety mode (Page 196)

5.8.2 Configuration in RUN im H-System (H-CiR)

H-CiR allows you to add new fail-safe I/O to your fault-tolerant system or delete existing fail-safe I/O from your system.

The procedure is similar to that described in section "Configuring F-I/O with CiR (Page 65)", taking into consideration information in the "Fault-Tolerant Systems S7-400H (<http://support.automation.siemens.com/WW/view/en/82478488>)" manual.

Note

Parameter reassignment of fail-safe I/O is not supported.

Access Protection

6.1 Overview of access protection

Purpose and mode of operation

Access protection protects S7 F/FH Systems from unauthorized access, such as undesirable downloads to the F-CPU from the Engineering System (ES). In addition to the password for the F-CPU, you need an additional password for the safety program for S7 F/FH Systems.

The table below provides information about the password for the F-CPU and the password for the safety program.

| Password for F-CPU | |
|-----------------------|--|
| Password assignment | In <i>HW Config</i> during configuration of the F-CPU in the "Protection" tab of the "Properties" dialog box |
| Password request when | <ul style="list-style-type: none"> • Downloading the entire S7-program from the <i>CFC Editor</i> or <i>SIMATIC Manager</i> • Downloading safety program changes from the <i>CFC Editor</i> • Performing a memory reset from the <i>CFC Editor</i> or <i>SIMATIC Manager</i> • Changing non-interconnected inputs in CFC test mode |
| Password validity | <p>Access permission is valid without restriction until it is explicitly canceled using the corresponding function of <i>SIMATIC Manager</i> (PLC > Access Rights > Cancel menu command) or you close the last <i>STEP 7</i> application.</p> <p>Access permission can become invalid if the hardware configuration of the CPU is changed and downloaded.</p> |

| Password for safety program | |
|------------------------------------|---|
| Password assignment | In <i>SIMATIC Manager</i> , Options > Edit safety program menu. |
| Password request when | <ul style="list-style-type: none"> • Compiling changes to the safety program • Downloading changes to the safety program • Disabling and enabling safety mode • Changing non-interconnected inputs in CFC test mode • Saving the safety program as a reference • Changing the shutdown behavior in the "Safety Program" dialog • Adding F-I/O in which safety mode has been enabled or that only support safety mode • Opening the Properties dialog for F-I/O in <i>HW Config</i> • Making changes in the PROFIsafe tab in <i>HW Config</i> • Making changes in the "F-Configuration" tab for a fail-safe intelligent DP slave <p>In addition, starting from <i>PCS 7V7.1</i>:</p> <ul style="list-style-type: none"> • Opening an F-Chart • With an open F-Chart <ul style="list-style-type: none"> – Editing object properties of an F-block – Assigning parameters to an input/output on an F block – Instantiating an F-block – Inserting an F block or CFC chart • With F-Runtime groups <ul style="list-style-type: none"> – Opening a CFC – Opening an F-Runtime group in the runtime view – Moving an F-Runtime group in the runtime view – Modifying the properties of an F-Runtime group <p>Starting from <i>PCS 7V8.2</i>, the following applies:</p> <ul style="list-style-type: none"> • The password prompt is omitted when the CFC chart with fail-safe component is opened. • The password prompt occurs only in the case of safety-related changes, i.e. when the signature of the safety program changes. <p>For this reason, changes can now be made in the standard program of a CFC chart with fail-safe components without a password prompt.</p> <ul style="list-style-type: none"> • The password prompt occurs independent of whether the user creates or changes an F-block explicitly or this happens implicitly, e.g. by a copy operation. |
| Password validity | The access permission lasts for one hour after correct password entry, during which time it is reset to another hour after each action requiring a password, or until access permission is explicitly revoked in <i>SIMATIC Manager</i> (Options > Edit safety program menu command, then click the Password button followed by the Cancel access rights button). |

6.2 Setting up access rights for the F-CPU

Procedure

1. Select the F-CPU or its S7 program in *SIMATIC Manager*.
2. Select the **PLC > Access Rights > Setup** menu command. On the "Protection" tab of the displayed dialog, enter the password that was assigned during parameter assignment of the F-CPU.

Access permission is valid until you explicitly revoke it again (**PLC > Access Rights > Cancel**) or close the last *STEP 7* application.

WARNING

Limiting access using the ES

If you have not activated access protection to limit access to the Engineering System to persons authorized to modify safety programs, you must use the following organizational measures to ensure the effectiveness of the password protection:

- Only authorized persons may have access to the password.
- Authorized persons must explicitly cancel the access permission for the F-CPU before leaving the ES. If you do not implement this measure consistently, you must additionally use a screen saver whose password can only be accessed by authorized persons.

In safety mode, access authorization by means of the F-CPU password must not be active when changes are made to the standard user program, because the safety program can then also be changed. To rule out this possibility, you must configure protection level "1".

If only one person is authorized to change the standard user program and the safety program, protection level "2" or "3" should be configured to ensure that other persons have only limited access or no access to the standard user program and safety program.

If safety mode is active after access permission is revoked, check to determine whether

- the collective signature of the safety program online
and
- the collective signature of the accepted safety program are identical.

If not, download the correct safety program to the F-CPU again.

Note

Automatic downloading of safety programs is not supported in multiprojects. The passwords must be entered at the time of downloading to the respective F-CPU.

Transferring the safety program to multiple F-CPUs

WARNING

Transferring the safety program to multiple F-CPUs

If multiple F-CPUs can be reached from an ES via a network (e.g. MPI), you must take the following additional measures to ensure that the safety program is downloaded to the correct F-CPU.

Use F-CPU-specific passwords, e.g. a password for the F-CPUs with appended MPI address "FCPUPW_8". The password has a maximum of 8 characters, including at least one special character. In STEP 7 V5.5.4 HF9 and higher, the password must contain 8 characters for new projects.

Note the following:

- Before a safety program for which access permission by means of an F-CPU password does not yet exist is downloaded to an F-CPU, any existing access permission for another F-CPU must first be canceled.

Changing the password

A password can only be changed by reconfiguring.

In the S7 F System, you must switch the F-CPU to STOP for this.

In the S7 FH-System, a password change (configuration change) is possible without a process interruption (in RUN).

WARNING

Password protection

After a cold restart, the current password is deleted from the RAM load memory and the old password from the flash EPROM memory card becomes valid again. To prevent too many people from knowing the old password on the flash EPROM memory card, you should take organizational measures.

6.3 Setting up access permission for the safety program

Setting up/changing an access permission for the safety program

Criteria for a secure password

To ensure a secure password, it must meet the following criteria when created for the first time or changed:

- Password length: at least 8, maximum of 32 characters
- At least one upper case letter of the Latin alphabet (A - Z); also diacritical marks (umlauts and letters with accents)
- At least one lower case letter of the Latin alphabet (a - z); also ß and diacritical marks (umlauts and letters with accents)
- At least one number (0-9)
- At least one of the following special characters:

~ ! @ # \$ % ^ & * _ - + = ` | \ () { } [] : ; ' " < > , . ? /

These criteria apply when the "Increased password security" option is activated in the "Create password for safety program" dialog.

Requirement

To set up an access permission for the safety program, a safety program (F-chart) must exist.

Procedure


To set up or change the password for the safety program, follow these steps:

1. Select the F-CPU or its S7 program in SIMATIC Manager.
2. Select the menu command **Options > Edit safety program**.
3. Click the "Password" button in the displayed "Safety Program" dialog. Perform the step required for your situation:
 - During the initial setup of a new password, select the password in conformance with the criteria described below and enter it in the "New password" and "Reenter password" fields. In this case, the "Old password" field is deactivated.

By selecting the "Increased password security" check box, you can use a more secure password that conforms to the description "Criteria for a secure password" above. You can find additional information on this in section ""Password for Safety Program Creation" dialog (Page 182)".

6.3 Setting up access permission for the safety program

- To change a password, you must enter the old password in the "Old password" field. Then, choose the new password and enter it in the "New password" and "Reenter password" fields.
When the "Increased password security" check box is selected, the description "Criteria for a secure password" above applies to the password selection.
- You can use the "Logout" button in the "Access permission" area to revoke the 1-hour access permission period since the last time the password was entered. Any user who then wants to perform an action that requires entry of a password must now enter the password for the safety program again.

| |
|--|
|  WARNING |
| Limiting access using the ES If you have not activated access protection to limit access to the Engineering System to persons authorized to modify safety programs, you must use the following organizational measures to ensure the effectiveness of the password protection: <ul style="list-style-type: none">• Only authorized persons may have access to the password.• Authorized persons must explicitly cancel the access permission for the safety program before leaving the ES. If you do not implement this measure consistently, you must additionally use a screen saver whose password can only be accessed by authorized persons. |

Note

The access permission relates to the safety program itself and not the persons that work on the ES. This must be taken into consideration, particularly in relation to multi-user engineering projects.

Note

Automatic editing and compiling of safety programs is not supported.
The password must be valid during the respective action.

Assigning a new password for the safety program

If a password has not yet been created for the safety program, you will be prompted to do so if a password is required for the desired configuring task, e.g. when inserting an F-block in a CFC chart or when inserting fail-safe modules in HW Config.

You can find additional information on the password prompt in section "Overview of access protection (Page 69)" in the "Password for safety program" table.

 **WARNING****Passwords must be unique**

To improve access protection, use different passwords for the F-CPU and the safety program.

The passwords of various safety programs must also be different.

Changing the password for the safety program

You change the password by entering the old password and then entering the new password twice.

Canceling access permission for the safety program

You can revoke the access permission at any time using the password for the safety program. Follow the procedure below:

1. Select the F-CPU or its S7 program in SIMATIC Manager.
2. Select the menu command **Options > Edit safety program**.
3. Click the "Password" button in the displayed dialog.
4. In the "Create password for safety program" dialog, click the "Logout" button in the "Access permission" area.

Programming

7.1 Overview of programming

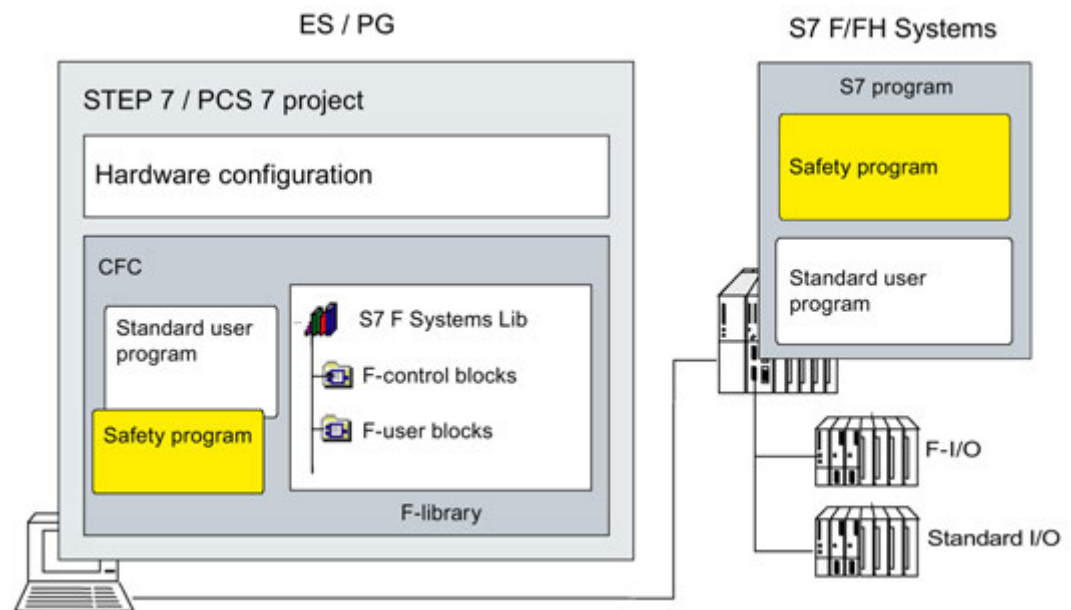
Introduction

A safety program consists of F-blocks that you select from the F-library and interconnect in the "CFC Editor" and F-blocks that are automatically added when the safety program is generated.

When the safety program is generated, fault-control measures are automatically added to the safety program you created and additional safety-related checks are performed.

Schematic structure of a project with standard user program and safety program

The figure below shows the schematic structure of an S7 program in the ES and the F-CPU:



The S7 program typically consists of a standard user program in which you program the parts of the program not required for the safety function and a safety program for the safety function.

7.1.1 Structure of the safety program

Representation of the program structure

The following figure shows the schematic structure of a safety program for *S7 F Systems*. A safety program consists of CFC charts with F-blocks that are assigned to F-runtime groups.

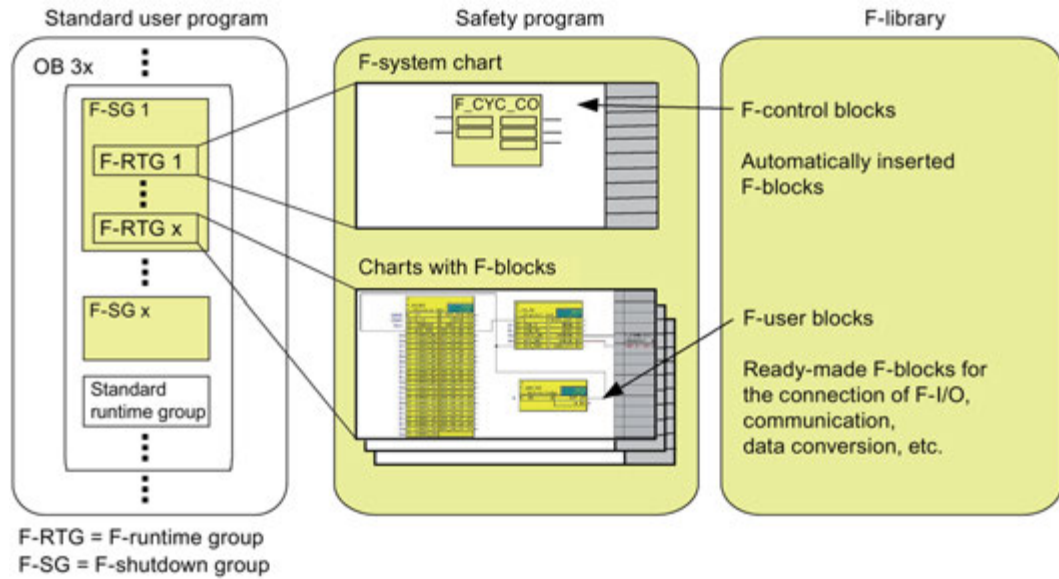


Figure 7-1 Components of the safety program in *S7 F Systems*

Explanation of the program structure

The safety program contains F-runtime groups and charts assigned to them. The charts contain F-blocks including their parameter assignment and interconnection.

F-runtime groups are combined into F-shutdown groups.

The F-shutdown groups are inserted at the start of a cyclic interrupt OB (OB 30 to OB 38).

The cyclic interrupt OB can also contain standard runtime groups.

Support for creating the program structure

In CFC V8.2 and higher, there is the so-called chart-based runtime group management in which the blocks of a CFC chart are automatically managed in a chart-oriented manner in their own runtime groups. The "Chart-based insertion" option must be activated in the properties of the chart folder or the CFC chart for this.

If a CFC chart with F-blocks is integrated in the chart-based runtime group management, a runtime group is created automatically not only for blocks in the so-called standard program but also for the contained F-blocks, the safety program.

- In this case, it is no longer necessary to manually move the F-blocks to their own runtime group.
- The name of the runtime group for the F-blocks contains the name of the CFC chart with the addition of "_F".

You will find further information on this in the Process Control System PCS 7, CFC for SIMATIC S7 (<https://support.industry.siemens.com/cs/ww/en/view/90683154>) manual, "Special features of F-blocks in CFC charts".

F-runtime groups

You are not permitted to insert F-blocks directly in tasks (OBs) when programming the safety program.

A runtime group becomes an F-runtime group only when it is called in its F-blocks. If no F-blocks are contained in the runtime group, it is regarded as a standard runtime group.

Your safety program consists of multiple F-runtime groups.

In CFC V8.2 and higher, the creation of F-runtime groups by CFC is supported. Note the paragraph "Support in creating the program structure" regarding this.

F-shutdown groups

An F-shutdown group is a self-contained unit of your safety program. An F-shutdown group contains the user logic which is simultaneously executed or shut down.

The F-shutdown group contains one or more F-runtime groups that are assigned to a common task. You can select whether a fault during execution of the safety program is to trigger a full shutdown of the entire safety program or a partial shutdown, that is, shutdown only of the F-runtime group in which the fault occurred.

F-blocks can exchange data between F-shutdown groups only via special F-blocks. All F-channel drivers belonging to an F-I/O must be located in the same F-shutdown group.

See also

Creating the Safety Program (Page 80)

F-STOP (Page 93)

7.2 Creating the Safety Program

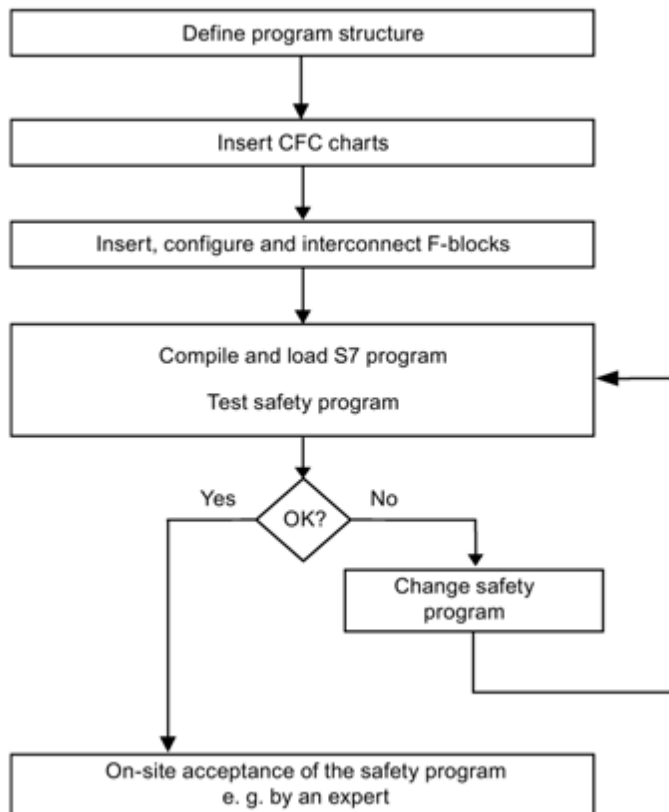
7.2.1 Basic procedure for creating the safety program

Requirements

- You must create a project structure in *SIMATIC Manager*.
- You must already have configured the hardware components of your project - in particular, the F-CPU and the F-I/O - prior to programming the safety mode.
- You must have assigned your safety program to an F-capable central processing unit, such as a CPU 410-5H.

Basic procedure

Proceed as follows to create a safety program:



7.2.2 Defining the program structure

Introduction

When designing an S7 program for S7 F/FH Systems, you must answer the following additional questions as compared to a standard program:

- Which components of the S7 program must be fail-safe?
- What response times do you want to achieve?

Based on this, you must divide your S7 program into different OB 3x cyclic interrupts.

Note

You can improve performance by writing sections of the program that are not required for the safety functions in the standard user program.

When determining which elements to include in the standard user program and which to include in the safety program, keep in mind that the standard user program can be modified and downloaded to the F-CPU more easily. In general, changes in the standard user program do not require an acceptance test.

Rules for the program structure


You must keep the following rules in mind when designing a safety program for S7 F/FH Systems:

- You can only assign F-Shutdown groups with F-Blocks to the OB 3x (OB 30 to OB 38) cyclic interrupts.
- A chart can contain both F-Blocks and standard blocks. You cannot compile these charts as F-Block types.
- The F-I/O can only be accessed in the safety program via the F-Channel drivers.

7.2.3 Assigning parameters for the maximum F-cycle monitoring

The F-CPU monitors the F-Cycle time for each cyclic interrupt OB 3x that contains F- Runtime groups. The first time you compile the S7 program, you will be prompted to enter a value for the maximum cycle time "MAX_CYC" that can elapse between two calls of this OB. For information about setting F-Monitoring times, refer to chapter "Run times, F-Monitoring times, and response times (Page 460)".


If you need to change the maximum F-Cycle time, set the F-Cycle time at the MAX_CYC parameter of block F_CYC_CO-OB3x in chart @F_CycCo-OB3x.


| |
|--|
|  WARNING |
| Default setting of the maximum MAX_CYC |
| The default setting for the maximum F-Cycle time is 3,000 milliseconds. Check whether this setting is appropriate for your process. Change the default setting if necessary. |

Note

For changes to the F-Cycle time during RUN mode, refer to chapter " Changing the time ratios or F-Monitoring times (Page 207) ".

7.2.4 Rules for programming

| |
|--|
|  WARNING |
| Do not change values created during compilation |
| During compilation, you must not change automatically executed placements, interconnections, and parameter assignments of F-Blocks. |
| <ul style="list-style-type: none">• In particular, you must not manipulate the structural components COMPLEM and PAR_ID of F-Data types.• You must not change the F-Control blocks that are automatically inserted in the safety program (in F-System charts) (except the MAX_CYC parameter at F_CYC_CO).• In F-Blocks, you are only permitted to interconnect or assign the parameters that are described in the online help or manual. |
| You must not change or delete the F-Blocks in the block container. |

| |
|---|
|  WARNING |
| The cyclic interrupt OB 3x group call interval is monitored relative to the maximum value; that is, monitoring is performed to determine whether the call is executed often enough, but not whether it is executed too often. |
| For this reason, you must implement fail-safe times using F-Blocks such as F_TON, F_TOF, and F_TP, rather than via counters (OB calls). |

7.2.5 Notes for working with CFC

 **WARNING**

Compression changes the signature

Compressing a CFC program (using the **Options > Customize > Compile/Download** menu command in the *CFC Editor*), changes the collective signature of your safety program.

You must therefore do this prior to the acceptance test.

F-Blocks appear in the CFC chart highlighted in color. They are highlighted in yellow to indicate that a safety program is involved.

CFC charts and F-Runtime groups with F-Blocks are yellow and marked with an "F" in order to distinguish them from the charts and runtime groups of the standard user program.

7.2.6 Inserting CFC charts

Procedure

In the charts folder, insert individual CFC charts in the same way as for standard user programs:

- In *SIMATIC Manager* by selecting the menu command **Insert > S7 Software > CFC**
- Directly in the *CFC Editor* with the menu command **Chart > New**

Note

To install the newly inserted CFC charts directly in the respective planned cyclic interrupt OB 3x, you must position the CFC installation pointer accordingly.

Hierarchical charts

Chart outputs of a lower-level chart that are not interconnected internally cannot be further interconnected in the higher-level chart.

7.2.7 Inserting F-Runtime groups

Rules for F-runtime groups of the safety program

- To achieve F-cycles whose length are as equal as possible, we recommend the following procedure:

If you mix F-runtime groups and standard runtime groups in a cyclic interrupt OB, execute the F-runtime groups *before* the standard runtime groups. Otherwise, the runtime of the F-shutdown group is extended unnecessarily, thus affecting the response time.

- An F-runtime group must retain the presetting for the reduction ratio and phase offset runtime properties as follows:
 - Reduction ratio = 1
 - Phase offset = 0

You are not permitted to change these values.

- You are not permitted to move the automatically generated F-runtime groups. You are also not permitted to make any changes within this F-runtime group.

| |
|--|
|  WARNING |
|--|

| |
|--|
| Optimization of the run sequence in the <i>CFC</i> can alter the collective signature and make the response times of the safety program worse. |
|--|

| |
|---|
| Optimization of the run sequence is therefore not possible. |
|---|

Procedure

You insert F-runtime groups in the runtime editor of the CFC Editor in the same way as for standard user programs.

7.2.8 F-Shutdown groups

Rules for F-Shutdown groups of the safety program

- You are not permitted to interconnect F-Blocks belonging to different F-Shutdown groups.

For more information, refer to chapter "Programming data exchange between F-Shutdown groups in an F-CPU (Page 101)"

- All F-Channel drivers that belong to an F-I/O must be located in the same F-Shutdown group.

Defining F-Shutdown groups

As soon as you place F-Blocks in the *CFC Editor* for the first time, all F-Runtime groups in each OB 3x form an F-Shutdown group.

Dividing/combining F-Shutdown groups through manual placement of F_PSG_M

When you add or delete one or more F_PSG_M blocks in your project, the order of your F-Shutdown groups will change. If you change the layout of your F-Shutdown groups, you must make sure that the F-module driver and all associated F-Channel drivers are integrated in the same F-Shutdown group.

You have the option of dividing one F-Shutdown group into two F-Shutdown groups. To do so, in the runtime editor of the *CFC Editor*, place the F_PSG_M block in the last F-Runtime group to be associated with the first F-Shutdown group. All subsequent F-Runtime groups then form the second F-Shutdown group. The F_PSG_M block is not an F-Block. However, you are still permitted to place it in F-Runtime groups. For more information, refer to chapter "Determining the runtime sequence (Page 87)".

The number of F-Shutdown groups in all tasks is limited to 110. The number of F-Runtime groups in one task is unlimited.

You have the option of combining two F-Shutdown groups. To do so, in the runtime editor of the *CFC Editor*, delete the F_PSG_M block between the F-Shutdown groups. If you combine F-Shutdown groups that exchange data by means of F-System blocks into one common F-Shutdown group, you must delete these F-System blocks and replace them with direct interconnections.

7.3 Inserting and interconnecting F-Blocks

7.3.1 Inserting F-Blocks

Procedure

Insert the F-Blocks into your chart as usual in *CFC*.

Note

All F-Blocks are highlighted in yellow in the *CFC Editor* and in *SIMATIC Manager*. Only these blocks are part of your safety program. In addition, the F-User Blocks folder in the F-Library contains standard blocks, for example, to convert F-Data types to standard data types.

Rules for F-Blocks

- The blocks in the **F-Control Blocks** folder are automatically inserted when the S7 program is compiled. You are not permitted to insert these blocks yourself.
- You are not permitted to place an F-Block instance in several F-Runtime groups. This can take place, for example, by copying and inserting an F-Runtime group into another task.

Note

F-libraries in different versions

Your ES can contain *multiple* versions of the F-Library at the same time. However, a safety program can only contain F-Blocks from *one* version.

| |
|--|
|  WARNING |
|--|

| |
|---|
| Entries for F-Blocks in the symbol table must not be changed |
|---|

| |
|---|
| You are not permitted to change or delete the names of the F-Blocks in the "Symbol" column of the symbol table in your S7 program. This also applies for changes in the symbol table that is assigned to the F-Library. |
|---|

7.3.2 Parameter assignment and interconnection of F-Blocks


Procedure

Inputs and outputs of the F-Blocks are parameterized and interconnected as usual in *CFC*.

Rules for parameter assignment and interconnection of F-Blocks

- You are only permitted to assign parameters or interconnect the parameters documented in chapter "F-libraries (Page 229)".
- You are not permitted to interconnect input EN and output ENO of the F-Blocks and F-Runtime groups. Likewise, you are not permitted to assign a value of 0 (FALSE) to EN.
- The F-Data types are implemented in the program as structures in which only the first **DATA** component is relevant for you.

If you do not take this into account, the safety program/F-Runtime group will go into F-STOP, i.e., an F-Startup will be required.

| |
|--|
|  WARNING |
| <p>Illegal changes to input parameters of F-Blocks can cause a shutdown of the safety program and its outputs.</p> <p>Changes can be made to the input parameters of F-Blocks with F-Data types as follows:</p> <ul style="list-style-type: none"> • Offline using the <i>CFC Editor</i> <li style="text-align: center;"><i>or</i> • Online using the CFC test module with safety mode disabled <p>If you change F-Data types online with safety mode enabled without using the CFC test mode, this can cause a shutdown of the relevant outputs or an F-STOP will be initiated.</p> |

Recommendation: meaningful names for placed F-Blocks

Assign a meaningful name to each placed F-Block. You are free to choose the name.

7.3.3 Determining the runtime sequence

Correct runtime sequence of F-Blocks

The sequence of the F-Blocks within the F-Shutdown group is relevant. The number of F-Runtime groups into which the F-Shutdown group is divided is irrelevant.

In principle, the correct runtime sequence of the different F-Block types is as follows:

1. Automatically placed:
 - F-Module drivers for F-I/O with inputs or with inputs and outputs
 - F-Communication blocks and F-System blocks for receiving
 - F-Blocks for converting data
2. F-Channel drivers for inputs
3. F-Blocks for user logic

4. F-Channel drivers for outputs
5. Automatically placed:
 - F-Block F_PLK
 - F-Block F_PSG_M
 - F-Module drivers for F-I/O with outputs or with inputs and outputs
 - F-Communication blocks and F-System blocks for sending
 - F-Block F_PLK_O
 - F-Block F_DIAG

The runtime sequence of the F-Blocks listed in Items 1 and 5 is corrected automatically when the S7 program is compiled. You must, however, always ensure that the F-Channel drivers and F-Blocks for user logic are placed appropriately and adhere to the sequence described above. This ensures that all inputs are read first, the appropriate processing steps are initiated, and then all outputs are written to.

Determining the runtime sequence

You determine the runtime sequence in the *CFC Editor* in the same way as for a standard user program.

Note

A change in the runtime sequence also changes the collective signature.

7.4 Automatically inserted F-Blocks

F-Control blocks

During compilation of a CFC chart with F-Blocks, the following F-Control blocks are automatically inserted into the safety program:

- F_DIAG
- F_CYC_CO
- F_PLK
- F_PLK_O
- F_PS_12
- F_PS_MIX
- F_PSG_M *
- F_TEST
- F_TESTC
- F_TESTM

*) The F_PSG_M block is only placed one time during the migration of *Failsafe Blocks (V1_1)* or for programs with *Failsafe Blocks (V1_2)* of *S7 F Systems V5.2* with no SP.

During compilation of a CFC chart with F-Blocks, the following blocks are automatically inserted into the standard user program:

- DB_INIT
- DB_RES
- F_SHUTDN
- RTGLOGIC
- F_VFSTP1
- F_VFSTP2
- F_MOVRWS *
- F_CHG_WS *

*) Insertion of the blocks F_MOVRWS and F_CHG_WS depends on your programmed user logic.

WARNING

Do not change automatically inserted F-Control blocks

The automatically inserted F-Control blocks are visible following compilation. You must not delete these F-Blocks and must not make any changes to them, as this could cause errors during the next compilation. For exceptions, refer to the description of the F-Blocks in Appendix "F-libraries (Page 229)".

Note

When the S7 program is compiled, additional blocks (DB_RES) and calls that you are not permitted to change are automatically inserted at the beginning of the runtime sequence in OB 100.

7.5 F-Startup and reprogramming restart/startup protection

F-startup

S7 F Systems does not distinguish between a cold restart and warm restart of the CPU. Exception: F-blocks F_CHG_BO, F_CHG_R, F_MOV_R, F_SWC_CB and F_SWC_CR. For more information, refer to sections "Blocks and F-Blocks for data conversion (Page 274)" and "Multiplex blocks (Page 399)". Both a cold restart and a warm restart of the CPU results in an F-startup.

Note

Startup type "Cold restart"

In PCS 7 and when blocks from PCS 7 libraries are used, the startup type "Cold restart" is not permitted.

After an F-startup, the safety program starts up automatically with the initial values.

An F-startup takes place:

- After a CPU STOP when you perform a warm restart or a cold restart of the F-CPU.
- After an F-STOP when you perform the following steps:
 - Set the value "1" at the "Restart" input for the restart.
 - After you accept the value, reset it back to the original value "0".

After a partial shutdown of the safety program, only the F-shutdown groups that were in F-STOP perform an F-startup.

F-shutdown groups that are not fault-free remain in F-STOP.

WARNING

Saved error information is lost following an F-startup.

After a STOP of the F-CPU, the F-system automatically reintegrates the F-I/O following an F-startup.

A data handling error or an internal fault can also trigger a safety program restart with the initial values of the F-blocks. If your process does not allow such a startup, you must program a restart/startup protection in the safety program: Process data outputs must be blocked until manually enabled. Enabling of the process data output must not occur until it is safe to do so and faults have been corrected.

One of the following actions is required after troubleshooting:

- User acknowledgement at the F-channel driver
- User acknowledgement at F-block F_RCVBO or F_RCVR, or F_RDS_BO

For F-blocks F_R_BO and F_R_R, which are used for data exchange between F-runtime groups, the reintegration of the receive data occurs automatically.

Restart/startup protection

If the process does not permit automatic startup of the safety program with initial values, you must program a reaction to the F-startup. The F-block F_START is available for signaling an F-startup of the safety program with initial values.

The COLDSTRT output parameter signals the occurrence of an F-startup.

Examples

You can use the following measures to react to a startup of the safety program with initial values:

- Programming of an **interlock** of the outputs after startup using the PASS_ON passivation inputs at the channel drivers for outputs. To do this, interconnect the COLDSTRT output of the F-block F_START with the S input of an SR-Flip-Flop (F_SR_FF) and the Q output of the F_SR_FF with PASS_ON of the F-channel driver for outputs. You can then enable the interlock manually:
 - Using a button that is queried via an F-I/O.
 - or
 - Through an input on the ES/OS via the F-block F_QUITES. In *S7 F Systems* V6.2 and higher, also via SWC_QOS (F_SWC_BO).

You must interconnect the Q output of the F-channel driver belonging to the button or the OUT output of F_QUITES or F_SWC_BO with the R input of F_SR_FF.
- Programming of an **idle loop** so that the internal states of the safety program correspond to the process state again.
- Programming using multiplexers: The output of a multiplexer F_MUX2_R is controlled by the COLDSTRT output of the F-block F_START. As a result, a different program branch can be executed after a startup than in cyclic operation.

7.6 F-STOP

Introduction

If the safety program detects a safety-related fault, a fault reaction is triggered. If no fail-safe values can be output, the fault reaction that is then carried out is called an F-STOP.

Types of F-STOP

There are two types of F-STOP:

- **Full shutdown**

All F-shutdown groups of the F-CPU are shut down. The shutdown is carried out in the following order:

- Initially, the F-shutdown group in which the fault was detected is shut down.
- All other F-shutdown groups are then shut down within a period of time equal to twice the F-monitoring time you assigned for the slowest OB.

- **Partial shutdown**

Only the F-blocks of the F-shutdown group in which a fault was detected are shut down.

A shutdown of F-shutdown groups means:

- The outputs of the F-I/O controlled by the F-shutdown group are passivated.
- The F-channel drivers of the F-shutdown group set the outputs QBAD to "1" and QUALITY to "0".
- The safety-related communication of the F-shutdown group with other F-CPU's is interrupted.
- The data exchange of the F-shutdown group with other F-shutdown groups is interrupted.
- In the case of data exchange from the safety program to the standard user program, the last valid values are provided to the standard user program.
- The F_SHUTDN block generates a message you can display on an OS.
- Diagnostic events are entered in the diagnostics buffer of the F-CPU.

The standard user program of the F-CPU continues running even after an F-STOP.

In order to assign the F-STOP parameters, use the "Shutdown behavior" button in the "Safety Program" dialog. See also " "Shutdown Behavior" dialog box (Page 181) ".

Faults that trigger an F-STOP

- Falsification of
 - Data
 - Program flow
 - Code
- CPU fault

Faults that always trigger a full shutdown

A full shutdown is triggered following an OB request error (e.g. due to an OB overload), regardless of the F-STOP parameter assignment.

Manual triggering of an F-STOP

You can manually trigger an F-STOP by creating a positive edge at the RQ_FULL input of the F-block "F_SHUTDOWN".

Sequence of an F-STOP in S7 FH Systems

Before a safety program on a redundant F-CPU goes to F-STOP, it performs the following steps:

- The fault occurs in the master:
 - The S7 FH-System performs a master/standby changeover.
 - The F-CPU that was previously the master then switches to TROUBLESHOOTING operating state.

If a fault is not subsequently found, the F-CPU is reconnected. You will find more information in the " Fault-Tolerant Systems S7-400H (<http://support.automation.siemens.com/WW/view/en/82478488>) " manual.

If a fault was found, the F-CPU that was previously the master switches to DEFECT operating state.

In the case of redundant F-CPU's, faults on one side do not trigger a shutdown of the program execution.

- The fault occurs in both F-CPU's:
 - The safety program goes to F-STOP immediately.

Ending an F-STOP

Perform an F-startup as described in section " F-Startup and reprogramming restart/startup protection (Page 91) ".

See also

Initial run and startup characteristics (Page 209)

Group passivation (Page 114)

7.7 Creating F-Block types

7.7.1 Introduction

S7 F Systems gives you the option of generating an F-Block type from the CFC chart of a safety program. You can re-use F-Block types in other safety programs.

7.7.2 Rules for F-Block types

Rules for F block types

When creating a new F block type with F blocks, you follow the same basic procedure as for the standard user program. The same rules apply as for creating block types in *CFC*. In addition, you must also keep the following in mind:

- The new F block type can only contain F blocks from the F-Library, except for:
 - F-Channel driver
 - F blocks for F-Communication
 - F blocks F_CHG_BO, F_CHG_R, F_MOV_R or F_SWC_x
 - All F-Control blocks
For more information, refer to the section "S7 F Systems Lib V1_3 SP2 F-control blocks (Page 414)".
 - All F-System blocks, except for F_START
For more information, refer to the section "F-System blocks (Page 367)".
- The F blocks that are called in the new F block type and the F blocks of the entire safety program in which the F block type is used must originate from the same library version. F blocks from different versions of the F-Library are not permitted.
- You are not permitted to connect an output of the F block with two chart inputs/outputs.
- The runtime sequence within one F block type is not automatically corrected during compilation. The sequence determined during creation is retained.

Note

If the runtime sequence is different from the data flow, for example, due to feedback, compilation of the F block type is canceled with an error.

- The chart inputs/outputs of the new F block type can have both F-Data types and standard data types.

- You are not permitted to use names of F blocks in the F-Library as the names of F block types.
- For instances of F blocks that are called in an F block type, we recommend that you assign names as follows:
 - Numbers only, as specified in the *CFC Editor*
or
Alphanumeric names, but that must begin with F_
 - Upper-case letters only
 - No "_" at the end

Note

Starting with *S7 F Systems* V6.1, you can set the *S7_m_c* attribute to 'true' for standard outputs. If you use this option, however, your safety program will no longer be backward-compatible with *S7 F Systems* V6.0.



WARNING

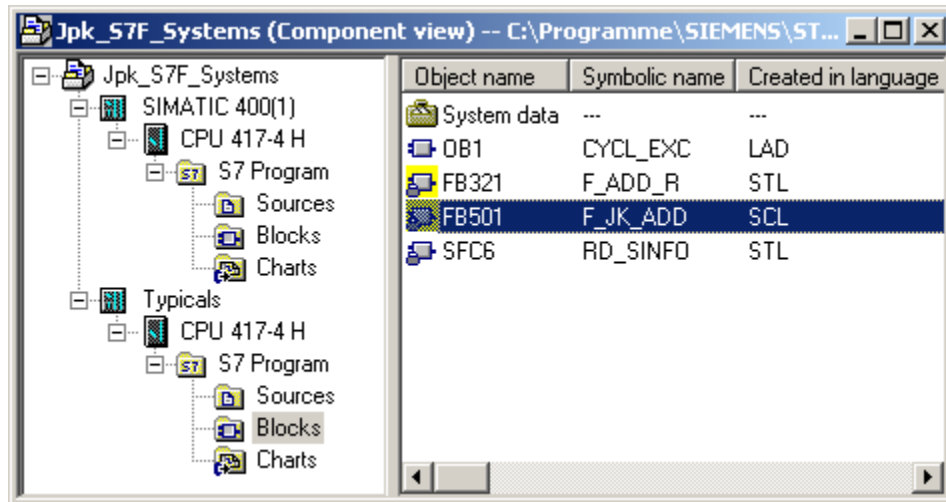
Outputs of F blocks always use the predefined initial values

When creating F block types, you are not permitted to change any initial values at F block outputs. *CFC* allows this and shows you the change. However, *S7 F Systems* always uses the initial values described in the F block description under "Default".

7.7.3 Creating F-Block types with "Compile Chart as F-Block Type"

Procedure

1. Create the CFC chart in a separate S7 program that is assigned to an F-CPU. The S7 program can be located in the same project.



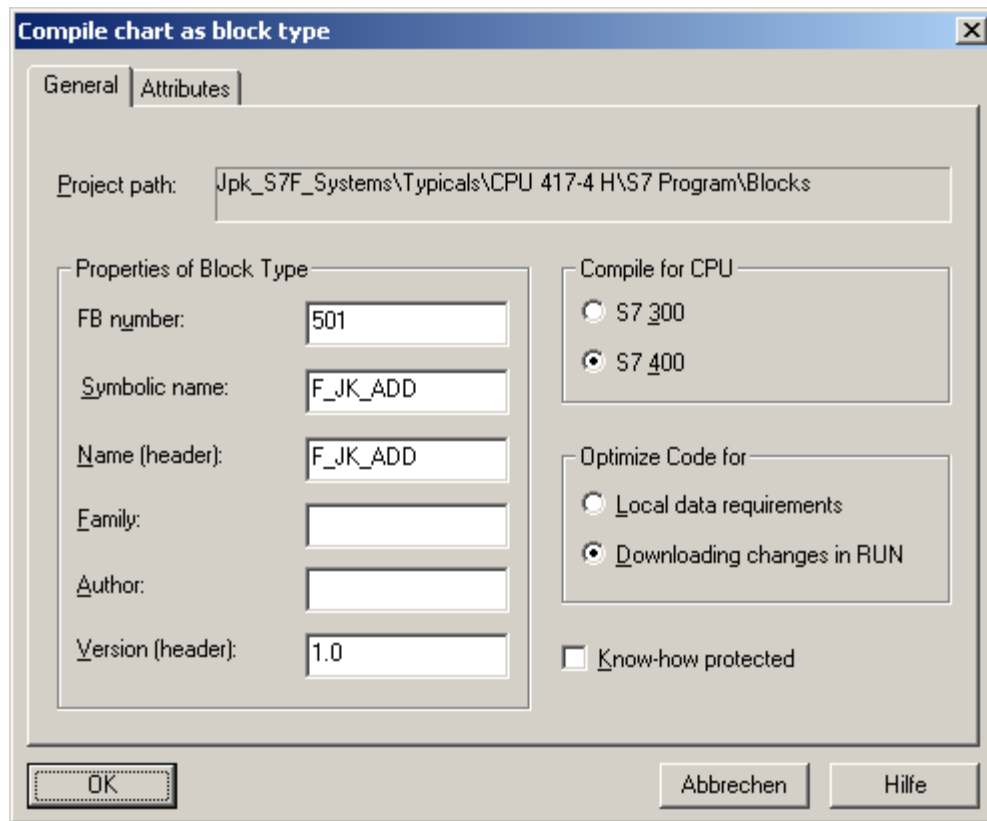
Note

Use a separate AS station to create an F-block type!

Always use a separate AS station that contains only the safety program of the F-block type to create an F-block type.

2. Open the desired chart.

3. Select the menu command **Chart > Compile > Chart as block type**. A dialog for entering the block properties is displayed.



4. Open the properties of the new F-block type.

Ensure that the names under "Symbolic name" and "Name (Header)" are identical.

When the FB number is assigned, it must be ensured that the selected number does not fall within a number range of the libraries used in the project. This is necessary to prevent conflicts.

5. Activate the "Compile for CPU - S7 400" and "Optimize code for - Download changes in RUN" options and confirm with OK.

The know-how protection is always activated independent of the setting of the option.

Result: A new F-block is generated that you can use in a safety program.

6. Insert the new F-block type together with the F-blocks that it calls in a safety program and test it there.

Note

Attributes

Attributes whose name begins with "F_" are managed by *S7 F Systems*. Assign other names for your own attributes to prevent them from being deleted or overwritten during compiling.

7.7.4 Modifying F-Block types

Modifying F-Block types

You have to update modified F-Block types just as with all other block types in the *CFC-editor*. To do this, open the dialog "Block types" with the menu command **Options > Block Types** and click the button "New Version".

Modifications to already used F-Block types may result in your having to recompile and download the complete S7 program afterwards.

If you want to use a new version of the F-Library, you have to compile the F-Block types with this new version of the F-Library. You can find additional information in the chapter "Updating custom F-block types (Page 43)"

See also

Downloading changes (Page 203)

System Acceptance Test (Page 213)

7.7.5 Integrating F-parameters of custom F-block types in the printout of the safety program

Requirement

You have already created an F-block type and opened it as a CFC chart.

Procedure

1. Select the chart I/O that you want to include in the safety-related printout.
2. Select "Object properties" in the shortcut menu to open the object properties of the tag. Make sure that you open the object properties of the structure and not the subordinate elements.
3. Open the "Attributes" tab and enter the "F_PrintTypParam" attribute in an empty row. Set the value of the attribute to "TRUE".
4. Repeat this process for all chart I/Os you want to include in the safety-related printout.
5. Select the menu command **Chart > Compile > Chart as Block Type** and compile the F-block type.

Result

The I/Os with the "F_PrintTypParam" attribute appear in a printout of the F-program with the "Print safety-relevant parameters" option.

See also

Creating F-Block types with "Compile Chart as F-Block Type" (Page 97)

7.8 Programming data exchange between F-Shutdown groups in an F-CPU

Rules for the data exchange between F-shutdown groups

- If you want to exchange data between two F-shutdown groups, you are not permitted to directly interconnect the inputs and outputs. You must use special F-blocks for this.
- Information about the run sequence can be found in section " Determining the runtime sequence (Page 87) ".

Available F-blocks

You must use the following F-system blocks for the data exchange between F-blocks in different F-shutdown groups:

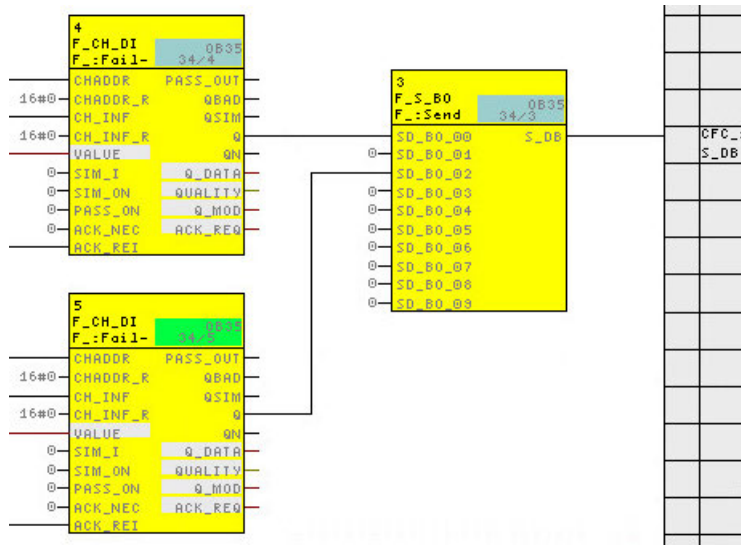
| F-block | Description |
|-----------------|--|
| F_S_R / F_R_R | Safe transmission of 5 data elements of F-data type F_REAL. |
| F_S_BO / F_R_BO | Safe transmission of 10 data elements of F-data type F_BOOL. |

Procedure

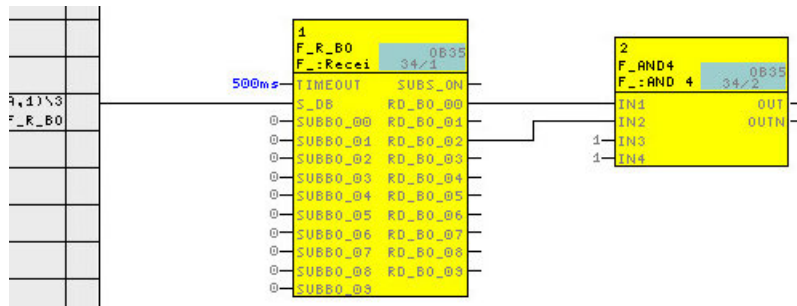
1. In the F-shutdown group *from* which data is to be transmitted, you insert an F-block of type F_S_R or F_S_BO.
2. In the F-shutdown group *to* which data is to be transmitted, you insert an F-block of type F_R_R or F_R_BO.
3. Interconnect inputs SD_R_xx of F_S_R or SD_BO_xx of F_S_BO with the data to be transmitted.
4. Interconnect outputs RD_R_xx of F_R_R or RD_BO_xx of F_R_BO with the inputs of the F-blocks for further processing of the received data.
5. Interconnect the S_DB output of the send block with the S_DB input of the associated receive block.
6. Set the desired F-monitoring time for the TIMEOUT inputs of the F_R_R and F_R_BO receive blocks.

For information regarding calculation of the F-monitoring time, see section "Run times, F-Monitoring times, and response times (Page 460)".

Examples: Excerpt from the chart of the F-shutdown group *from* which data will be transmitted



Example: Excerpt from the chart of the F-shutdown group *to* which data will be transmitted



Note

If you interconnect F-blocks in different F-shutdown groups directly (without the above-indicated F-system blocks), a compilation error will be generated at the next compilation.

If you interconnect F-blocks within a F-shutdown group with the above-indicated F-system blocks, an error message will be generated.

7.9 Data exchange between safety program and standard user program

Overview

The standard user program and safety program use different data formats. Safety-related F-Data types are used in safety programs. Standard data types are used in the standard user program.

Therefore, you have to use special conversion blocks for data exchange.

Parameters are output as safety-related F-Data types in the safety program.

Data Transfer from the Safety Program to the Standard User Program

If data from the safety program is to be processed further in the standard user program, e.g., for monitoring, then you have to insert a data conversion block (*F_F data type_data type*) in the *CFC editor* between the two programs to convert F-Data types to standard data types. You can find these blocks in the F-Library.

Data Transfer from the Standard User Program to the Safety Program

Data from the standard user program cannot be processed in the safety program until a validity check is performed. You must perform additional process-specific validity checks in the safety program to ensure that no hazardous conditions can arise.

To process data from the standard user program in the safety program, you have to use F-Blocks to convert the data (*F_data type_F data type*) from the standard data types to safety-related F-Data types. If necessary, you must then subject the converted data to a programmed validity check. These F-Blocks can be found in the F-Library.

7.9.1 Programming data exchange from the safety program to the standard user program

Available conversion blocks

The following blocks are available for conversion:

| Block | Description |
|----------|----------------------------------|
| F_FBO_BO | Converts F_BOOL to standard BOOL |
| F_FR_R | Converts F_REAL to standard REAL |
| F_FI_I | Converts F_INT to standard INT |
| F_FTI_TI | Converts F_TIME to standard TIME |

Procedure

Proceed as follows:

1. Insert blocks of type F_FBO_BO, F_FR_R, F_FI_I, or F_FTI_TI into the charts of the standard user program. You can find these blocks in the F-Library.
2. Interconnect the inputs of type F_data type with similar signals from the safety program.
3. Interconnect the outputs of the standard data type with similar signals from the standard user program.

7.9.2 Programming data exchange from the standard user program to the safety program

Available F-conversion blocks

The following F-Blocks are available for conversion:

| F-Block | Description |
|----------|----------------------------------|
| F_BO_FBO | Converts standard BOOL to F_BOOL |
| F_I_FI | Converts standard INT to F_INT |
| F_R_FR | Converts standard REAL to F_REAL |
| F_TI_FTI | Converts standard TIME to F_TIME |

Procedure

Proceed as follows:

1. Insert F-Blocks of type F_BO_FBO, F_I_FI, F_TI_FTI, or F_R_FR into the charts of the safety program.
2. Interconnect the inputs of the standard data type with similar signals from the standard user program.
3. Interconnect the outputs of F-Data types by means of a validity check with similar signals in the safety program.

Note

The adding, changing, and deleting of interconnections from the standard user program to the F-Conversion blocks is considered a change in the safety program, even if this involves interconnections of a standard data type. This means that access permission is required for compilation (see "Access Protection (Page 69)").

WARNING

Validity check

The F-Blocks F_BO_FBO, F_I_FI, F_TI_FTI, and F_R_FR only perform a data conversion. This means you must program additional measures for validity checks in the safety program.

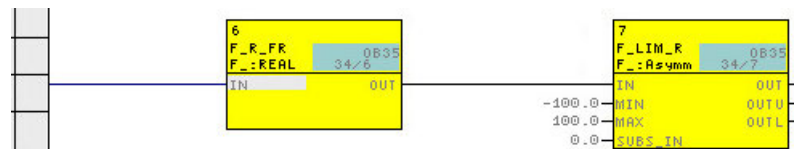
Validity check

The simplest type of validity check is a range definition with a fixed upper and lower limit, such as F_LIM_R.

Not all input parameters can be checked for plausibility in a sufficiently simple manner.

Example: Converting standard data types to F-Data types

Section from an F-Chart for converting REAL to F_REAL:



7.10 Implementation of user acknowledgment

Options for user acknowledgment

You can implement a user acknowledgment by means of the following:

- An acknowledgment button that you connect to an F-I/O with inputs
- A manual input via the OS

User acknowledgement via acknowledgment button

Note

When a user acknowledgment is implemented using an acknowledgment button and a communication error, F-I/O fault or channel fault occurs on the F-I/O to which the acknowledgment button is connected, an acknowledgment for reintegration of these F-I/O is also no longer possible. This "blocking" can only be canceled by a STOP/RUN transition of the F-CPU. For this reason, provision for an additional acknowledgment via an OS is recommended for the acknowledgment for reintegration of an F-I/O to which an acknowledgment button is connected.

User acknowledgement via an OS

For a user acknowledgement via the OS, the "Fail-safe acknowledgement" operator function based on "Secure Write Command++" can be configured and executed starting from *S7 F Systems V6.2* and *S7 F Systems Lib V1_3 SP2*. You can find additional information on this in section "Application case: Fail-safe acknowledgement (Page 138)".

The following description shows the configuration prior to *S7 F Systems V6.2*. Existing configurations of this type can continue to be used.

User acknowledgement via OS with the F-block F_QUITES

For implementation of a user acknowledgment via an OS, the F-block F_QUITES is required.

Procedure for programming the user acknowledgement via an OS

1. Insert the F-block F_QUITES in your safety program. The acknowledgment signal is available for the user acknowledgments at the OUT output of F_QUITES for evaluation.
2. Set up a field on your OS for manual input of the "Acknowledgement value" "6" (1st acknowledgment step) and "Acknowledgement value" "9" (2nd acknowledgment step) at the IN input of F_QUITES.

- Optional: Evaluate the Q output of F_QUITES on your OS in order to display the time window within which the 2nd acknowledgment step has to occur or to indicate that the 1st acknowledgment step has already been completed.

 WARNING

The two acknowledgment steps must not be triggered by a single operation, e.g. by entering the acknowledgment steps including time conditions automatically in a program and triggering by a single operation! The two separate acknowledgment steps will also prevent erroneous tripping of an acknowledgment by your non-fail-safe OS.

 WARNING

If access to multiple F-CPU's that use F_QUITES for fail-safe acknowledgment is possible from your OS, or if you have internetworked operator control and monitoring systems and F-CPU's (with F_QUITES F-blocks), you must be confident that the intended F-CPU is actually being referenced before executing the two acknowledgment steps:

- For this purpose, store a designation that is unique network-wide for the F-CPU in a DB of your standard user program in every F-CPU.
- Set up a field on your OS from which you can read out the designation of the F-CPU online from the DB before executing the two acknowledgment steps.
- Optional: Set up a field on your OS in which the designation of the F-CPU is additionally permanently stored. Then, a simple comparison of the designation of the F-CPU that is read out online with the permanently stored designation allows you to see whether the intended F-CPU is being referenced.

See also

F-Blocks for F-Communication between F-CPU's (Page 237)

F-Channel drivers for F-I/O (Page 310)

F-I/O access

Access via F-driver blocks

The following are required for each F-I/O:

- One F-module driver (to be generated by the compiler)
- One F-channel driver for each utilized input/output channel of the F-I/O

In S7 F/FH Systems, the F-I/O is accessed via F-module drivers. These F-module drivers communicate with the F-modules via direct I/O access. For this reason, a process image partition for the F-module does not have to be configured.

Cyclic updating of the process image is not required.

Note

When a process image partition is configured for the F-module, insertion of the F-module and call-up of the F-driver block in different cyclic interrupt OBs are not permitted!

Failure to observe this may cause sporadic data falsifications during communication with the F-module.

F-module drivers

The F-module driver undertakes the PROFIsafe communication between the safety program and the F-I/O. The F-module driver is automatically placed and interconnected in the safety program by the CFC driver generator.

F-channel drivers

The F-channel drivers in your safety program form the interface to a channel of an F-I/O and perform signal processing. There are different F-channel drivers depending on the F-I/O (see section F-Channel drivers for F-I/O (Page 310)).

You must place and interconnect F-channel drivers in the safety program.

For redundantly configured F-I/O, you need only one F-channel driver for two redundant channels

8.1 Positioning, interconnecting, and assigning parameters to F-Channel drivers

Requirement: Symbolic names

Enter a symbolic name (symbol) for each utilized channel in the symbol table. You must assign this symbol to the VALUE or I_OUT_D I/O of the associated F-channel driver. For more clarity, you should also comment the unused channels as reserved channels or unused channels in the symbol table.

Procedure

1. Place the suitable F-channel driver for each utilized input/output channel.
2. For each F-channel driver, interconnect the VALUE or I_OUT_D I/O with the symbol of the associated channel. This step is required for all placed F-channel drivers. For redundantly configured F-I/O, interconnect the VALUE I/O with the symbol of the channel with the lower channel address.
3. Interconnect the following inputs/outputs with your user logic:
 - The I inputs of the F-channel drivers F_CH_DO, F_CH_BO
 - The Q or QN outputs of the F-channel drivers F_CH_DI, F_PA_DI, F_CH_BI
 - The V outputs of the F-channel drivers F_CH_AI, F_PA_AI
4. Optional: Interconnect the simulation I/O.
5. Optional: Interconnect the PASS_ON input if you want to activate passivation of the channel, e.g. dependent on certain states in your safety program.
6. Optional: Assign the value "1" to the respective ACK_NEC input if a user acknowledgment is required for reintegration of the channel. The ACK_NEC input is assigned the value "0" by default (see section "Group passivation (Page 114)").
7. Interconnect the respective ACK_REI input with the signal for acknowledgment of reintegration (see section "Group passivation (Page 114)").
8. Optional: Interconnect the PASS_OUT or QBAD output in order to observe whether a fail-safe value or a valid process value is being output.
9. Optional: Evaluate the QUALITY output in the standard user program or on the OS if you want to query or observe the value status (quality code) of the process value.
10. Optional: Evaluate the ACK_REQ output in the standard user program or on the OS in order to determine whether a user acknowledgment is required.

Depending on the F-channel driver, there are other inputs and outputs you can or must interconnect (see appendix "F-Channel drivers for F-I/O (Page 310)")

8.2 Generating F-Module drivers

Generating F-module drivers

Use the driver generator of the *CFC* for this.

When the S7 program is compiled in the "Compile program" dialog, the "Generate module drivers" option is activated by default. Check the setting of the option, and activate the option if it is not activated.

When this option is activated, the driver generator places all automatically generated F-module drivers in their own CFC charts, named @F_(1), @F_(2), etc. The instances of the F-module drivers are automatically assigned the name that you have entered for the associated F-I/O in *HW Config* (F_Name_x). The F-channel drivers are interconnected with the associated F-module drivers.

If you are using *PCS 7*, additional blocks are also inserted by the driver generator (see *PCS 7* documentation).

8.3 Process data or fail-safe values

When are fail-safe values used?

The safety function requires that fail-safe values be used instead of process data for passivation of the entire F-I/O or individual channels of an F-I/O in the following cases:


- During an F-Startup
- When errors occur during safety-related communication (communication errors) between the F-CPU and F-I/O using the safety protocol in accordance with PROFIsafe
- If F-I/O or channel faults are detected (e.g. wire break, short-circuit, or discrepancy error)
- As long as you have enabled an F-I/O passivation on the F-Channel driver at input PASS_ON

Fail-safe output for F-I/O/channels of an F-I/O

In the case of an F-I/O with inputs, the F-System provides fail-safe values at the F-Channel driver during passivation instead of the process data pending at the fail-safe inputs.

A fail-safe value of 0 is provided for (digital) channels of data type BOOL.

For analog channels, you must assign the fail-safe values at input SUBS_V of the F-Channel driver and enable them by assigning 1 to input SUBS_ON or select the last valid value as the fail-safe value by assigning 0 (default value) to input SUBS_ON.

| |
|---|
|  WARNING |
| For F-I/O with inputs, the fail-safe value 0 provided at the F-Channel driver must be further processed for (digital) channels of data type BOOL in the safety program. |

In the case of an F-I/O with outputs, the F-System transfers fail-safe values to the fail-safe outputs during passivation instead of the output values provided by the F-Channel driver.

Reintegration

The changeover from fail-safe values to process data (reintegration of an F-I/O) is executed either automatically or after user acknowledgment on the F-Channel driver.

The reintegration method depends on the following:

- Cause of passivation of the F-I/O/channels of the F-I/O
- Parameter to be assigned by you on the F-Channel driver

Note

For F-I/O with outputs, acknowledgment after F-I/O faults or channel faults may only be possible minutes after the fault has been eliminated due to necessary test signal inputs (see F-I/O manuals).

See also

F-Channel drivers for F-I/O (Page 310)

8.4 Group passivation

Description

If you want to enable passivation of additional F-I/O when an F-I/O or a channel of an F-I/O is passivated by the F-System, you can use the PASS_OUT output or PASS_ON input to perform a group passivation of associated F-I/O.

Group passivation by means of PASS_OUT/PASS_ON can, for example, be used to force simultaneous reintegration of all F-I/O after startup of the F-System.

For group passivation, you must OR all PASS_OUT outputs of the F-Channel drivers in the group with F_OR4 F-Blocks and interconnect the result at the OUT output of F_OR4 with all PASS_ON inputs of the F-Channel drivers in the group.

See also

F-Channel drivers for F-I/O (Page 310)

Programming communication

9.1 Safety-related communication between F-CPU's

9.1.1 Configuring safety-related communication via S7 connections

Introduction

Safety-related communication between the safety programs of F-CPU's via S7 connections takes place by means of connection tables in *NetPro*, in the same way as with standard programs.

Note

In S7 F/FH Systems, safety-related communication via S7 connections is possible to and from the following F-CPU's:

- CPU 315F-2 and higher
 - CPU 317F-2 and higher
 - CPU 319F-3 and higher
 - CPU 412-4H and higher
 - CPU 414-4H and higher
 - CPU 414F-3 and higher
 - CPU 416F-x and higher
 - CPU 416-5H and higher
 - CPU 417-4H and higher
 - CPU 410-5H and higher
-

Note



CPU-CPU communication and public networks

Safety-related CPU-CPU communication is not permitted via public networks.

Creating an S7 connection in the connection table

For each communication connection between two F-CPU's, you must create an S7 connection in the connection table in *NetPro*.

9.1 Safety-related communication between F-CPU's

STEP 7 assigns a local ID and a partner ID for each connection end-point. If necessary, you can change the local ID in *NetPro*. You assign the local ID to the ID parameter of the appropriate F-blocks in the safety programs.

Note

Safety-related communication via S7 connections to unspecified partners is not possible.

Procedure for configuring S7 connections

You configure the S7 connections for safety-related CPU-to-CPU communication the same way as for standard systems, or even as a fault-tolerant S7 connection, if necessary.

Note

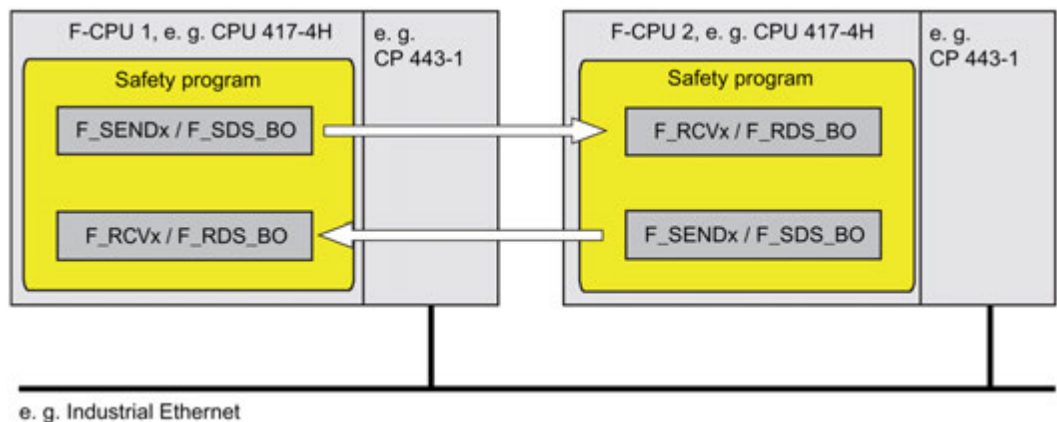
If you modify the configuration of S7 connections for safety-related communication, you must recompile the relevant S7 programs and download them to the F-CPU's.

Additional information

You will find a description of how to configure S7 connections in the following sources:

- Manual " Configuring Hardware and Communication Connections STEP 7 V5.x (<http://support.automation.siemens.com/WW/view/en/45531110>) "
- Manual " Automation System S7-400H Fault-tolerant Systems (<http://support.automation.siemens.com/WW/view/en/82478488>) "
- *STEP 7 online help*

9.1.2 Communication via F_SENDBO/F_RCVBO, F_SENDR/F_RCVR, and F_SDS_BO/F_RDS_BO



You use the **F_SENDBO/F_RCVBO**, **F_SENDR/F_RCVR**, and **F_SDS_BO/F_RDS_BO** F-Communication blocks for sending and receiving data in a fail-safe manner via S7 connections.

This allows you to safely transfer a *fixed* number of up to 20 data elements of F-Data type **F_REAL** and up to 20/32 data elements of F-Data type **F_BOOL**.

9.1.3 Programming safety-related CPU-to-CPU communication via S7 connections

Requirements for Programming

The following requirements must be met prior to programming:

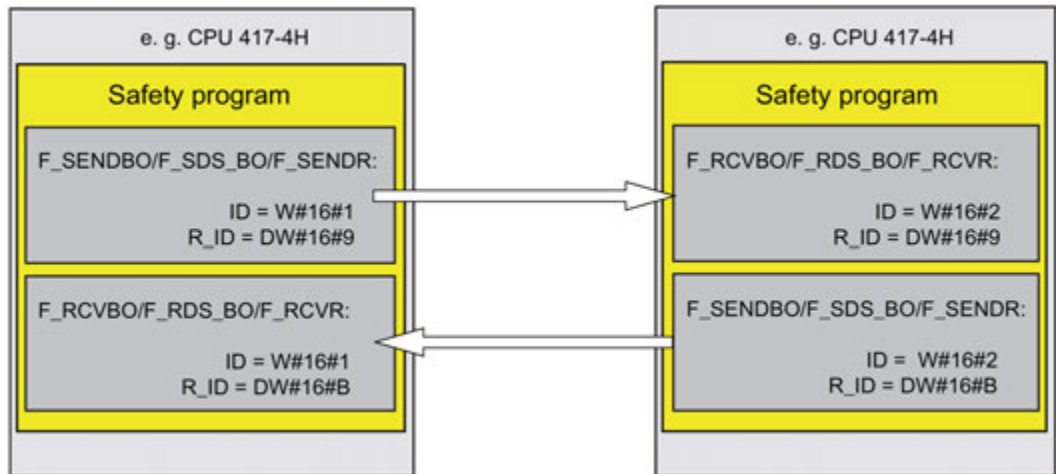
- The S7 connections between the relevant F-CPU's must be configured in *NetPro*
- Both CPU's must be configured as F-CPU's:
 - The "CPU contains safety program" option must be enabled
 - and*
 - The password for the F-CPU must be entered

Programming Procedure

1. In the safety program used to send data, insert the send F-Block **F_SENDBO/F_SDS_BO/F_SENDR**.
2. In the safety program used to receive data, insert the receive F-Block **F_RCVBO/F_RDS_BO/F_RCVR**.
3. Assign the local ID of the S7 connection (data type: WORD) configured in *NetPro* to the input ID of **F_SENDBO/F_SDS_BO/F_SENDR**.
4. Assign the local ID of the S7 connection (data type: WORD) configured in *NetPro* to the input ID of **F_RCVBO/F_RDS_BO/F_RCVR**.

9.1 Safety-related communication between F-CPU's

5. Assign an odd number (data type: DWORD) to the R_ID inputs of F_SENDBO/F_SDS_BO/F_SENDR and F_RCVBO/F_RDS_BO/F_RCVR. This defines an association between F_SENDBO/F_SDS_BO/F_SENDR and F_RCVBO/F_RDS_BO/F_RCVR. The associated F-Blocks are given the same R_ID value.



⚠ WARNING

The value for each address association (input parameter R_ID; data type: DWORD) is user-defined; however, it must be unique from all other safety-related communication connections in the network. The value R_ID + 1 is internally assigned and must not be used.

6. Interconnect inputs SD_BO_xx and SD_R_xx of the F-Blocks F_SENDBO/F_SDS_BO/F_SENDR with the send signals.
7. Interconnect the outputs RD_BO_xx and RD_R_xx of the F-Blocks F_RCVBO/F_RDS_BO/F_RCVR with the F-Blocks for further processing of the received signals.
8. Assign the fail-safe values to be made available at the outputs RD_BO_xx or RD_R_xx to the inputs SUBBO_xx and SUBR_xx of the F-Blocks F_RCVBO/F_RDS_BO/F_RCVR:
 - While the connection between communication peers is being established for the first time after F-Startup of the F-Systems
 - Whenever a communication error occurs
9. Assign the required F-monitoring time to the TIMEOUT inputs of F_SENDBO/F_SDS_BO/F_SENDR and F_RCVBO/F_RDS_BO/F_RCVR.

⚠ WARNING


It can be ensured (from a fail-safe standpoint) that a signal level to be transferred will be detected on the sender side and transferred to the receiver only if the signal is pending for at least as long as the assigned F-Monitoring time (TIMEOUT).


For information about calculating F-Monitoring times, refer to the section entitled "Run times, F-Monitoring times, and response times (Page 460)".

Note

For safety reasons, parameter assignment at the TIMEOUT inputs must take place within the minimum F-Monitoring time. TIMEOUT must not be used to increase availability.

10. To reduce the bus load, you can temporarily shut down communication between the F-CPU's by assigning "0" (default = "1") to input EN_SEND of F_SENDBO/F_SDS_BO/F_SENDR. In this case, send data are no longer sent to the associated F_RCVBO/F_RDS_BO/F_RCVR, and the recipient F_RCVBO/F_RDS_BO/F_RCVR provides the assigned fail-safe values for this time period. If communication was already established between the connection partners, a communication error is detected.
11. Optional: Evaluate the ACK_REQ output of F_RCVBO/F_RDS_BO/F_RCVR in the standard user program, for example, in order to query or to indicate whether user acknowledgment is required.
12. Interconnect the ACK_REI input of F_RCVBO/F_RDS_BO/F_RCVR with the reintegration acknowledgement signal.
13. Optional: Evaluate output SUBS_ON of F_RCVBO/F_RDS_BO/F_RCVR or F_SENDBO/F_SDS_BO/F_SENDR to query whether F_RCVBO/F_RDS_BO/F_RCVR is outputting the fail-safe values you assigned at inputs SUBBO_xx/SUBR_xx.
14. Optional: Evaluate the ERROR output of F_RCVBO/F_RDS_BO/F_RCVR or F_SENDBO/F_SDS_BO/F_SENDR in the standard user program, for example, in order to query or to indicate whether a communication error has occurred.
15. Optional: Evaluate output SENDMODE of F_RCVBO/F_RDS_BO/F_RCVR to query whether the F-CPU with the associated F_SENDBO/F_SDS_BO/F_SENDR is in deactivated safety mode.

| |
|--|
|  WARNING |
| If the F-CPU with the associated F_SENDBO/F_SDS_BO/F_SENDR is in deactivated safety mode, you can no longer assume that the data received from this F-CPU were generated safely. You must then implement organizational measures such as operation monitoring and manual safety shutdown to ensure safety in those portions of the system that are affected by the received data. Alternatively, you must output fail-safe values instead of the received data in the F-CPU with F_RCVBO/F_RDS_BO/F_RCVR by evaluating SENDMODE. |

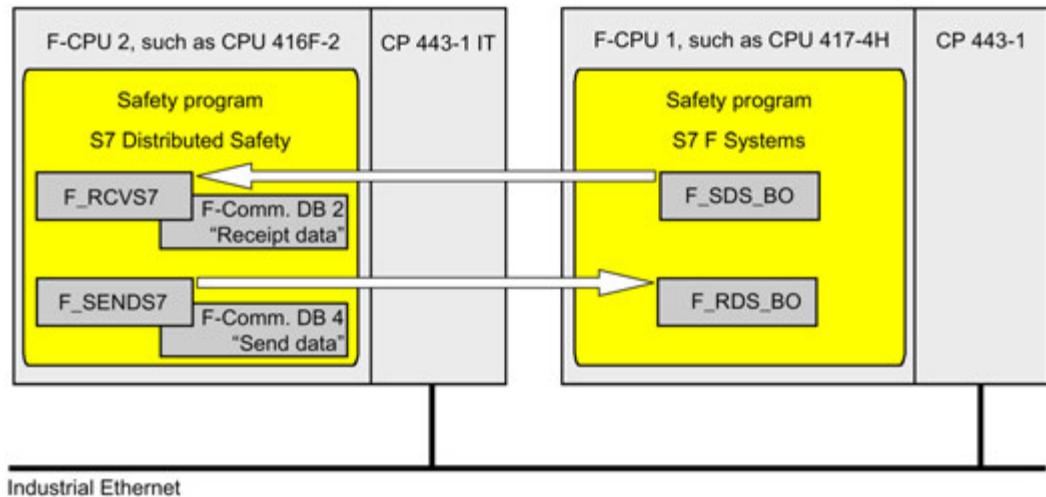
| |
|--|
|  WARNING |
| The S7 program must be recompiled if the S7 connections for communication between F-CPU's have been changed |
| If the safety program contains F-Blocks for safety-related CPU-to-CPU communication, the S7 program involved in communication must be recompiled after the following actions in order to update the connection data: |
| <ul style="list-style-type: none">• Copying of an F-CPU• Copying of a safety program or chart to another F-CPU• Changing of a communication peer of an S7 connection• Removal from/insertion into the multiproject of a project containing the communication peer of an S7 connection |

See also

Determining the runtime sequence (Page 87)

Safety engineering in SIMATIC S7 System Manual
(<http://support.automation.siemens.com/WW/view/en/12490443>)

9.2 Safety-related communication between S7 F Systems and S7 Distributed Safety



Procedure on the *S7 F Systems* side

On the *S7 F Systems* side, proceed as described in section "Safety-related communication between F-CPU's (Page 115)".

Particularity:

Communication between *S7 F Systems* and *S7 Distributed Safety* is only possible on the *S7 F Systems* side with the F blocks F_SDS_BO/F_RDS_BO.

Procedure on the *S7 Distributed Safety* side

On the *S7 Distributed Safety* side, proceed as described in section "Safety-related communication via S7 communications" in manual "S7 Distributed Safety - Configuring and Programming (<http://support.automation.siemens.com/WW/view/en/22099875>)".

Particularity:

For communication between *S7 F Systems* and *S7 Distributed Safety*, you must create the F-DB with exactly 32 data elements of data type BOOL on the *S7 Distributed Safety* side.

Operator inputs with the "Secure Write Command++" function

10

10.1 Concept of "Secure Write Command++"

Function

The "Secure Write Command++" functionality (SWC++) enables safety-related changes to be made to F-parameters in the safety program of an F-CPU from an operation station (OS).

The safety-related changes are executed by the following operator functions based on "SWC++":

- Maintenance Override
- Change process values
- Fail-safe acknowledgment

You can find additional information on the operator functions in section "Operator functions based on "Secure Write Command++" (Page 125)".

Note

The operator functions based on "SWC++" are available only under SIMATIC PCS 7.

With "SWC++" the actions for changing parameters in the F-CPU from the WinCC OS are separated into:

- | | |
|------------------------|---|
| Component in the F-CPU | <ul style="list-style-type: none"> • Transaction of the protocol • Receipt of parameters |
| Component in the OS | <ul style="list-style-type: none"> • Object that calculates the checksum • Check interface for confirming the transaction |

Each of these actions is carried out either in individual F-blocks in the F-CPU or individual objects in the OS.

Blocks for operator function based on "Secure Write Command++"

| Block type | Function | Name | Use *1) | Version |
|----------------------------|--|----------|---------------|---|
| Protocol block | Centralized control of operator input via the OS | F_SWC_P | CHG, MOS, QOS | Starting from <i>S7 F Systems</i> V6.1 and <i>S7 F Systems Lib</i> V1_3 |
| Parameter assignment block | Value change for data type F_BOOL | F_SWC_BO | MOS, QOS | Starting from <i>S7 F Systems</i> V6.1 and <i>S7 F Systems Lib</i> V1_3 |
| | Value change for data type F_REAL | F_SWC_R | MOS | Starting from <i>S7 F Systems</i> V6.1 and <i>S7 F Systems Lib</i> V1_3 |

10.1 Concept of "Secure Write Command++"

| Block type | Function | Name | Use *1) | Version |
|------------------------|--|----------|---------|---|
| | Value change for data type F_BOOL | F_SWC_CB | CHG | Starting from <i>S7 F Systems</i> V6.2 and <i>S7 F Systems Lib</i> V1_3 SP2 |
| | Value change for data type F_REAL | F_SWC_CR | CHG | Starting from <i>S7 F Systems</i> V6.2 and <i>S7 F Systems Lib</i> V1_3 SP2 |
| Operator control block | Establishes the connection to the WinCC faceplate. | SWC_MOS | MOS | Starting from <i>S7 F Systems</i> V6.1 and <i>S7 F Systems Lib</i> V1_3 |
| | Establishes the connection to the WinCC faceplate. | SWC_CHG | CHG | Starting from <i>S7 F Systems</i> V6.2 and <i>S7 F Systems Lib</i> V1_3 SP2 |
| | Establishes the connection to the WinCC faceplate. | SWC_QOS | QOS | Starting from <i>S7 F Systems</i> V6.2 and <i>S7 F Systems Lib</i> V1_3 SP2 |

- *1) Meaning of the abbreviations:
 "CHG" = Function "Change process values"
 "MOS" = Function "Maintenance Override"
 "QOS" = Function "Fail-safe acknowledgment"

Other components for operator functions based on "Secure Write Command++"

| Type | Description | Name | Use *1) | Version |
|------------------|---|--------|-----------------|--|
| "Chart-in-Chart" | Used for a time-controlled Maintenance Override | SWC_TR | MOS | Starting from <i>S7 F Systems</i> V6.1 and <i>S7 F Systems Lib</i> V1_3 |
| Faceplates | Faceplates for the OS | - | MOS CHG, QOS | Starting from <i>S7 F Systems</i> V6.1 Starting from <i>S7 F Systems</i> V6.2 |

- *1) Meaning of the abbreviations:
 "CHG" = Function "Change process values"
 "MOS" = Function "Maintenance Override"
 "QOS" = Function "Fail-safe acknowledgment"

Note

When used with PCS 7, one PO license is used for each instance of an operator control block in the safety program.

Operator input types

The action for safety-related changing of F-parameters in the safety program is referred to as an "operator input".

You perform an operator input in the OS via a faceplate. The operator input consists of a sequence of operations that can be performed by one or two operators.

See also

Blocks and F-Blocks for data conversion (Page 274)

10.2 Operator functions based on "Secure Write Command++"

Overview

The following operator functions are based on "Secure Write Command++" (SWC++):

- "Maintenance Override"

"Maintenance Override" allows you to set bypasses in the safety program from the OS.

Starting from *S7 F Systems* V6.1, you can create a bypass for up to three process signals for F_BOOL or F_REAL. The bypasses can be mutually interlocked, if required. In addition, you can use Maintenance Override to change fail-safe values for process signals and assign a reset time in order to reset the set bypasses automatically after this time.

- "Change process values"

"Change process values" allows you to change F-parameters in the safety program from the OS.

Starting from *S7 F Systems* V6.2 with *S7 F Systems Lib* V1_3 SP2, you can change an F parameter of data type F_BOOL or F_REAL with "SWC++" (F_SWC_CB / F_SWC_CR and SWC_CHG).

- "Fail-safe acknowledgment"

"Fail-safe acknowledgment" allows you to implement a fail-safe acknowledgment from the OS.

Starting from *S7 F Systems* V6.2 with *S7 F Systems Lib* V1_3 SP2, you can control reintegration of F-I/O via the ES/OS with "SWC++" (F_SWC_BO and SWC_QOS).

Note

Possible combinations of blocks

- The F_SWC_CB and F_SWC_CR blocks may only be used with SWC_CHG. It is not possible to use these blocks for SWC_MOS or SWC_QOS.
 - The F_SWC_BO block may only be used with SWC_MOS and SWC_QOS. It is not possible to use this block with SWC_CHG.
 - The F_SWC_R block may only be used with SWC_MOS. It is not possible to use this block with SWC_CHG.
-

10.3 Programming operator functions

10.3.1 Basic procedure

Basic procedure

To perform an operator function via an OS, follow these steps:

On the engineering station (ES)

1. Place an operator control block, e.g. SWC_QOS, one or more parameter assignment blocks, e.g. F_SWC_BO, and one or more protocol blocks F_SWC_P in your CFC chart and interconnect them.

For more information, refer to section "Placement, parameter assignment and interconnection of F-blocks in the CFC (Page 126)".

2. Configure the faceplate for the operator control block.

For more information, refer to section "Configuring the faceplate of the operator functions (Page 140)".

On the operator station (OS)

- Perform a value change with "Change process values" on an F-parameter.
- Create a bypass with "Maintenance Override" at the F-channel drivers and change the fail-safe value, if necessary.
- Perform a fail-safe acknowledgment with "Fail-safe acknowledgment".

You can find further information about these operator inputs on the OS in section "Executing operator functions (Page 145)".

10.3.2 Placement, parameter assignment and interconnection of F-blocks in the CFC

10.3.2.1 Introduction

Introduction

The following sections show you typical application cases for the individual operator functions.

You will be given information on the procedure for placement, parameter assignment and interconnection of blocks and F-blocks for the operator functions in CFC charts.

- "Change process values"
 - Application case: "Change process values" with logic blocks (Page 128)
 - Application case: "Change process values" with arithmetic block (Page 130)
- "Maintenance Override"
 - Application case: Simulating a F-channel driver (Page 132)
 - Application case: Grouped maintenance override with mutual interlock (Page 134)
 - Application case: Time-triggered maintenance override (Page 136)
- "Fail-safe acknowledgment"
 - Application case: Fail-safe acknowledgment (Page 138)

Note

The creation of F-block types based on the "Secure Write Command++" function is not supported.

Use of a keyswitch

To ensure that only authorized persons can perform operator inputs, you can connect the EN_SWC input of the F-block F_SWC_P to a keyswitch.

Input EN_SWC = 1 must be set during an operator input. When EN_SWC = 0 after an operator input, all existing bypasses are deactivated. However, set fail-safe values are retained.

 **WARNING**

The "Secure Write Command++" functionality allows changes to the safety program to be made during RUN mode.

As a result, the following additional safety measures are required:

- Ensure that operator inputs that could compromise plant safety cannot be carried out. You can use the EN_SWC input of the F-block F_SWC_P for this purpose, for example, by controlling it with a keyswitch or on a process-dependent basis via the safety program.
- Ensure that only authorized persons can carry out operator inputs.

Examples:

- Control the EN_SWC input of the F-block F_SWC_P with a keyswitch.
- Set up access protection for the operator stations where an operator function based on "Secure Write Command++" can be performed.

Multiple protocol blocks in a shutdown group

Starting from S7 F Systems V6.2 with S7 F Systems Lib V1_3 SP2, it is possible to place multiple protocol blocks for each shutdown group and to thus enable multiple simultaneous operator inputs from the OS.

This requires an interconnection between the ADR_OSPA output of the protocol block F_SWC_P and the ADR_SWC input of the associated operator control block.

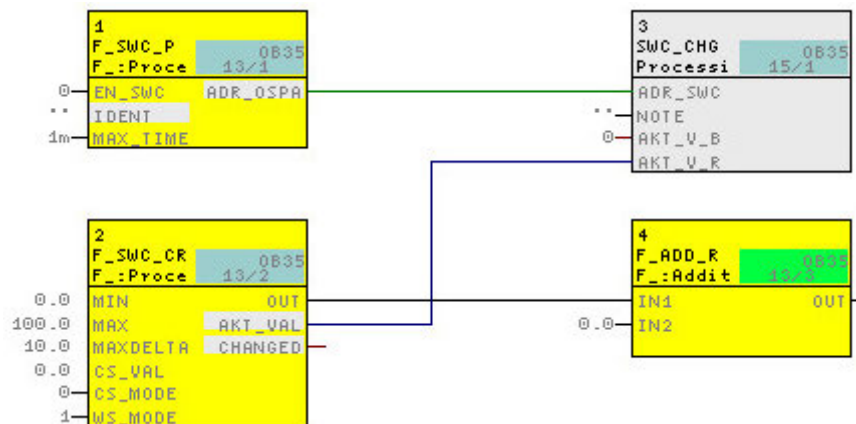
If multiple F_SWC_P blocks per shutdown group are present, the ADR_OSPA output of each F_SWC_P must be interconnected.

This interconnection is not required when simultaneous operator inputs per shutdown group are not needed.

Principle of configuration

1. Place the protocol block F_SWC_P and an operator control block, e.g. SWC_CHG, in a CFC chart.
2. Connect the ADR_OSPA output of F_SWC_P to the ADR_SWC input of the operator control block, e.g. SWC_CHG.

The following figure shows a possible configuration of the blocks and this interconnection.



The remaining configuration steps are described in the following application cases.

10.3.2.2 Application case: "Change process values" with logic blocks

Application

This application case shows you how to control a signal in your plant with the "Change process values" function dependent on a control signal from your plant.

Procedure

WARNING

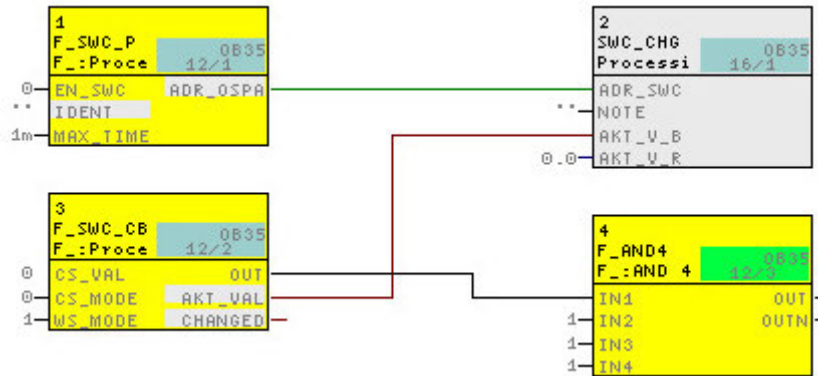
Warnings in the descriptions of the F-blocks

Observe the warnings in the descriptions of the F-block F_SWC_CB.

1. Place the SWC_CHG block in your CFC chart.
Observe the information on assigning names in section "SWC_CHG: Operator function for Change process values (Page 306)".
2. Place the F-block F_SWC_P, if necessary.
Optional:
If you place multiple protocol blocks F_SWC_P per shutdown group so that multiple simultaneous operator inputs are possible from the OS, connect the ADR_OSPA output of F_SWC_P to the ADR_SWC input of the associated operator control block SWC_CHG. This interconnection is not required when simultaneous operator inputs per shutdown group are not needed. You can find additional information on this in section "Introduction (Page 126)".
3. Place one F-block F_SWC_CB and F_AND4 each.
4. Connect the EN_SWC input of the F-block F_SWC_P to a keyswitch.
5. Assign the maximum duration of the operator input (default setting is 1 minute) at the MAX_TIME input of the F-block F_SWC_P.
6. Interconnect the inputs and outputs of the F-block F_SWC_CB:
Outputs:
 - OUT with the INx input of the F-block F_AND4
 - AKT_VAL with the AKT_V_B input of the SWC_CHG block
 Inputs:
 - Assign the initial value to the CS_VAL input that is to be transferred to the OUT output following a cold restart.
 - Optional: Assign the value "0" to the WS_MODE input if the value at the CS_VAL input is also to be transferred to the OUT output following a warm restart. The WS_MODE input is set to "1" by default.
7. Interconnect the INy input of the F-block F_AND4 with the controlling signal from your plant.
8. Interconnect the OUT output of the F-block F_AND4 with the signal of your plant to be controlled.
9. Before compiling, check the assignment of the SWC_CHG block. The block must be assigned to a standard runtime group.

10. Compile your CFC chart.

Additional connections between the SWC_CHG block, the F-blocks F_SWC_CB and F_SWC_P are created during compilation.




11. Follow the procedure as described in section "Configuring the faceplate of the operator functions (Page 140)".

10.3.2.3 Application case: "Change process values" with arithmetic block

Application

This application case shows you how to control a signal in your plant with the "Change process values" function dependent on a control signal from your plant.

Procedure

| | |
|---|---|
|  | <p>WARNING</p> <p>Warnings in the descriptions of the F-blocks</p> <p>Observe the warnings in the descriptions of the F-block F_SWC_CR.</p> |
|---|---|

1. Place the SWC_CHG block in your CFC chart.

Observe the information on assigning names in section "SWC_CHG: Operator function for Change process values (Page 306)".

2. Place the F-block F_SWC_P, if necessary.

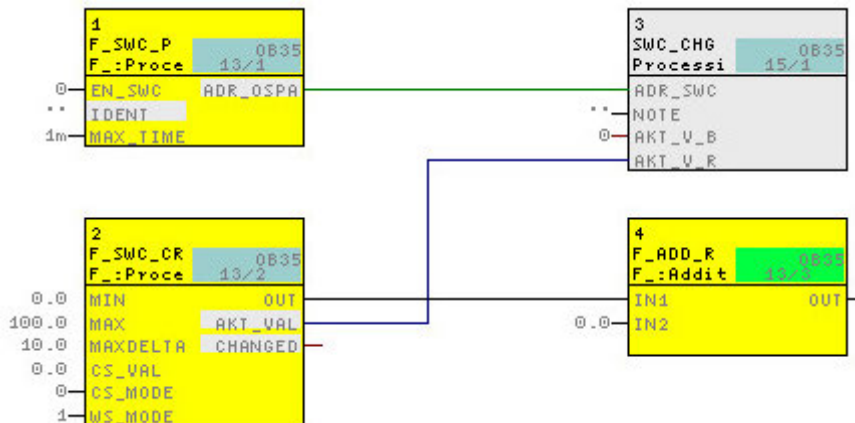
Optional:

If you place multiple protocol blocks F_SWC_P per shutdown group so that multiple simultaneous operator inputs are possible from the OS, connect the ADR_OSPA output of F_SWC_P to the ADR_SWC input of the associated operator control block SWC_CHG. This interconnection is not required when simultaneous operator inputs per shutdown group are not needed. You can find additional information on this in section "Introduction (Page 126)".

3. Place one F-block F_SWC_CR and F_ADD_R each.

4. Connect the EN_SWC input of the F-block F_SWC_P to a keyswitch.
5. Assign the maximum duration of the operator input (default setting is 1 minute) at the MAX_TIME input of the F-block F_SWC_P.
6. Interconnect the inputs and outputs of the F-block F_SWC_CR:
 - Outputs:
 - Interconnect OUT with the INx input of the F-block F_ADD_R
 - Interconnect AKT_VAL with the AKT_V_R input of the SWC_CHG block
 - Inputs:
 - Assign limits to the MIN and MAX inputs to specify the time during which the OUT output may be changed.
 - Assign the value of the maximum permissible increment of the change to the MAXDELTA input to specify the amount (+/-) by which the OUT output can change relative to the current value.
 - Assign the initial value to the CS_VAL input that is to be transferred to the OUT output following a cold restart.
 - Optional: Assign the value "0" to the WS_MODE input if the value at the CS_VAL input is also to be transferred to the OUT output following a warm restart. The WS_MODE input is set to "1" by default.
7. Interconnect the INy input of the F-block F_ADD_R with the controlling signal from your plant.
8. Interconnect the OUT output of the F-block F_ADD_R with the signal of your plant to be controlled.
9. Before compiling, check the assignment of the SWC_CHG block. The block must be assigned to a standard runtime group.
10. Compile your CFC chart.

Additional connections between the SWC_CHG block, the F-blocks F_SWC_CR and F_SWC_P are created during compilation.



11. Follow the procedure as described in section "Configuring the faceplate of the operator functions (Page 140)".

10.3.2.4 Application case: Simulating a F-channel driver

Application

This application case shows you how to simulate an F-channel driver with "Maintenance Override".

Procedure

 **WARNING**

Warnings in the descriptions of the F-blocks

Observe the warnings in the descriptions of the F-blocks F_SWC_BO / F_SWC_R.

1. Place the SWC_MOS block in your CFC chart.
Observe the information on assigning names in section "SWC_MOS: Command function for Maintenance Override (Page 307)".
2. Place the F-block F_SWC_P, if necessary.
Optional:
If you place multiple protocol blocks F_SWC_P per shutdown group so that multiple simultaneous operator inputs are possible from the OS, connect the ADR_OSPA output of F_SWC_P to the ADR_SWC input of the associated operator control block SWC_MOS. This interconnection is not required when simultaneous operator inputs per shutdown group are not needed. You can find additional information on this in section "Introduction (Page 126)".
3. Place an F-block F_SWC_BO that will start and stop the simulation.
4. Place an F-block F_SWC_BO or F_SWC_R that will change the simulation value, if such a change is desired.
5. Connect the EN_SWC input of the F-block F_SWC_P to a keyswitch.
6. Assign the maximum duration of the operator input (default setting is 1 minute) at the MAX_TIME input of the F-block F_SWC_P.
7. Connect the outputs of the F-block F_SWC_BO that will start and stop the simulation:
 - Connect OUT to the SIM_ON input of the F-channel driver
 - Connect AKT_VAL to the AKT_B1 input of the SWC_MOS block
8. Connect the outputs of the F-block F_SWC_BO or F_SWC_R that will change the simulation value:
 - Connect OUT to the SIM_I or SIM_V input of the F-channel driver
 - Connect AKT_VAL to the AKT_V_B or AKT_V_R input of the SWC_MOS block

9. Optional:

Assign a low limit and high limit for the fail-safe value (default setting 0.00 and 100.0, respectively) for the MIN and MAX inputs of the F-block F_SWC_R. Assign the CS_VAL input of the F-block F_SWC_R, if necessary.

10.Optional:

If you want to have the current value of an F-I/O displayed in the faceplate when a bypass is activated, connect the Q_MOD or V_MOD output of the F-channel driver to the V_MOD_B1B or V_MOD_B1R input of the SWC_MOS block.

11.Optional:

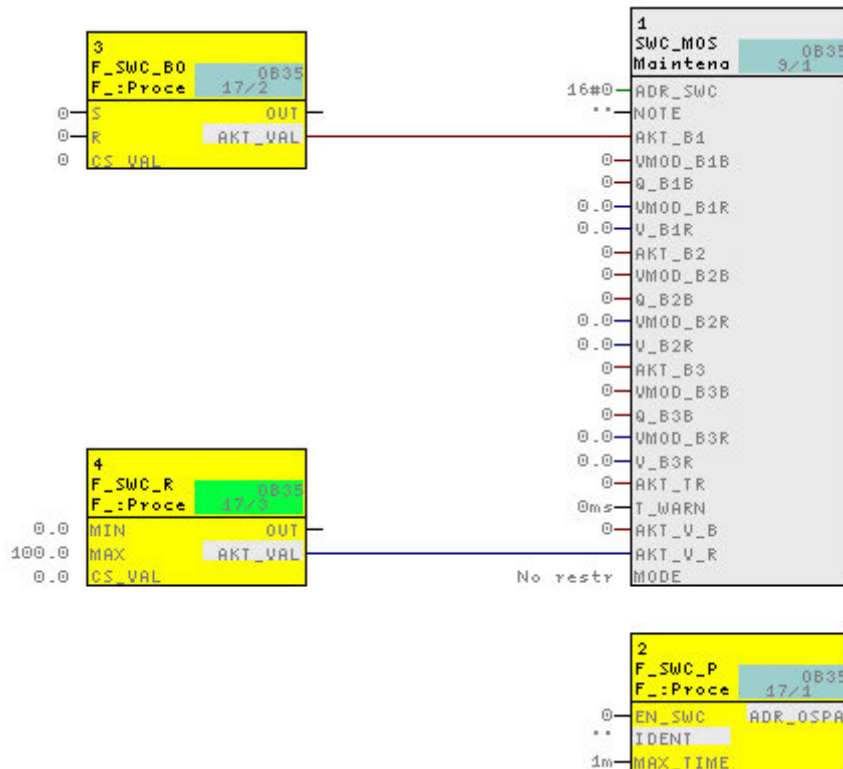
If you want to have the process value and its QUALITY displayed for the F-channel driver in the faceplate, connect the following outputs of the F-channel driver:

- Q_DATA or V_DATA output to the Q_BxB or V_BxR input of the SWC_MOS block.
- QUALITY output to the QUAL_Bx input of the SWC_MOS block

12.Before compiling, check the assignment of the SWC_MOS block. The block must be assigned to a standard runtime group.

13.Compile your CFC chart.

Additional connections between the SWC_MOS block, the F-blocks F_SWC_BO or F_SWC_R, F_SWC_P and the F-channel drivers are created during compilation.




14.Follow the procedure as described in section "Configuring the faceplate of the operator functions (Page 140)".

10.3.2.5 Application case: Grouped maintenance override with mutual interlock

Application

This application case shows you how to create a grouped "Maintenance Override".

Procedure

| |
|--|
|  WARNING |
| Warnings in the descriptions of the F-blocks |
| Observe the warnings in the descriptions of the F-blocks F_SWC_BO / F_SWC_R. |

1. Place the SWC_MOS block in your CFC chart.
Observe the information on assigning names in section "SWC_MOS: Command function for Maintenance Override (Page 307)".
2. Place the F-block F_SWC_P, if necessary.
Optional:
If you place multiple protocol blocks F_SWC_P per shutdown group so that multiple simultaneous operator inputs are possible from the OS, connect the ADR_OSPA output of F_SWC_P to the ADR_SWC input of the associated operator control block SWC_MOS. This interconnection is not required when simultaneous operator inputs per shutdown group are not needed. You can find additional information on this in section "Introduction (Page 126)".
3. Place 2 or 3 F-blocks F_SWC_BO that will start and stop the simulation.
4. If required, place an F-block F_SWC_BO or F_SWC_R that will change the simulation value.
5. Connect the EN_SWC input of the F-block F_SWC_P to a keyswitch.
6. Assign the maximum duration of the operator input (default setting is 1 minute) at the MAX_TIME input of the F-block F_SWC_P.
7. Connect the outputs of the F-blocks F_SWC_BO that will start and stop the simulation:
 - Connect OUT to the SIM_ON inputs of the associated F-channel drivers
 - Connect AKT_VAL to the AKT_Bx inputs of the SWC_MOS block
8. Connect the outputs of the F-block F_SWC_BO or F_SWC_R that will change the simulation value:
 - Connect OUT to the SIM_I and SIM_V inputs of the F-channel drivers
 - Connect AKT_VAL to the AKT_V_B or AKT_V_R input of the SWC_MOS block
9. Assign the MODE = 'MutualExclBypass' input of the SWC_MOS block in order to activate the mutual interlock.

10.Optional:

Assign a low limit and high limit for the fail-safe value (default setting 0.00 and 100.0, respectively) for the MIN and MAX inputs of the F-block F_SWC_R. Assign the CS_VAL input of the F-block F_SWC_R, if necessary.

11.Optional:

If you want to have the current value of an F-I/O displayed in the faceplate when a bypass is activated, connect the Q_MOD or V_MOD output of the F-channel driver to the V_MOD_BxB or V_MOD_BxR input of the SWC_MOS block.

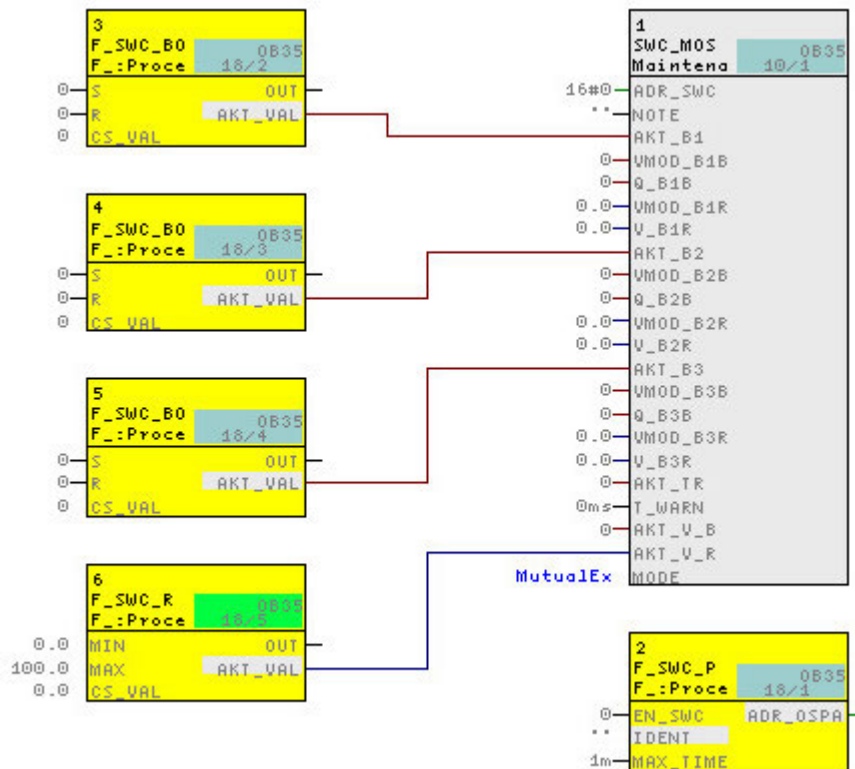
12.Optional:

If you want to have the process value and its QUALITY displayed for the F-channel driver in the faceplate, connect the following outputs of the F-channel driver:

- Q_DATA or V_DATA output to the Q_BxB or V_BxR input of the SWC_MOS block.
- QUALITY output to the QUAL_Bx input of the SWC_MOS block

13.Compile your CFC chart.

Additional connections between the SWC_MOS block, the F-blocks F_SWC_BO or F_SWC_R, F_SWC_P and the F-channel drivers are created during compilation.



14.Follow the procedure as described in section "Configuring the faceplate of the operator functions (Page 140)".

10.3.2.6 Application case: Time-triggered maintenance override

Application

This application case shows you how to create a time-controlled "Maintenance Override".

Procedure



WARNING

Warnings in the descriptions of the F-blocks

Observe the warnings in the descriptions of the F-blocks F_SWC_BO / F_SWC_R.

1. Place the SWC_MOS block in your CFC chart.
Observe the information on assigning names in section "SWC_MOS: Command function for Maintenance Override (Page 307)".
2. Place the F-block F_SWC_P, if necessary.
Optional:
If you place multiple protocol blocks F_SWC_P per shutdown group so that multiple simultaneous operator inputs are possible from the OS, connect the ADR_OSPA output of F_SWC_P to the ADR_SWC input of the associated operator control block SWC_MOS. This interconnection is not required when simultaneous operator inputs per shutdown group are not needed. You can find additional information on this in section "Introduction (Page 126)".
3. Place one or more F-blocks F_SWC_BO that will start and stop the simulation.
4. Place an F-block F_SWC_BO or F_SWC_R that will change the simulation value.
5. Place the Chart-in-Chart SWC_TR.
Assign the "reset time" (default setting = 0 ms) at the T_MAX input.
6. Connect the EN_SWC input of the F-block F_SWC_P to a keyswitch.
7. Assign the maximum duration of the operator input (default setting is 1 minute) at the MAX_TIME input of the F-block F_SWC_P.
8. Connect the outputs of the F-blocks F_SWC_BO that will start and stop the simulation:
 - Connect OUT to the SIM_ON inputs of the associated F-channel drivers
 - Connect AKT_VAL to the AKT_Bx inputs of the SWC_MOS block
9. Connect the outputs of the F-block F_SWC_BO or F_SWC_R that will change the simulation value:
 - Connect OUT to the SIM_I and SIM_V inputs of the F-channel drivers
 - Connect AKT_VAL to the AKT_V_B or AKT_V_R input of the SWC_MOS block

10. Connect the AKT_TR output of the "Chart-in-Chart" block SWC_TR to the AKT_TR input of the SWC_MOS block.

11. Optional:

Assign a low limit and high limit for the fail-safe value (default setting 0.00 and 100.0, respectively) for the MIN and MAX inputs of the F-block F_SWC_R. Assign the CS_VAL input of the F-block F_SWC_R, if necessary.

12. Optional:

Assign the prewarning time for the automatic reset of the active bypasses (default setting = 0 ms) at the T_WARN input of the SWC_MOS block.

13. Optional:

Set the MODE = 'MutualExclBypass' input of the SWC_MOS block in order to activate the mutual interlock.

14. Optional:

If you want to have the current value of an F-I/O displayed in the faceplate when a bypass is activated, connect the Q_MOD or V_MOD output of the F-channel driver to the V_MOD_BxB or V_MOD_BxR input of the SWC_MOS block.

15. Optional:

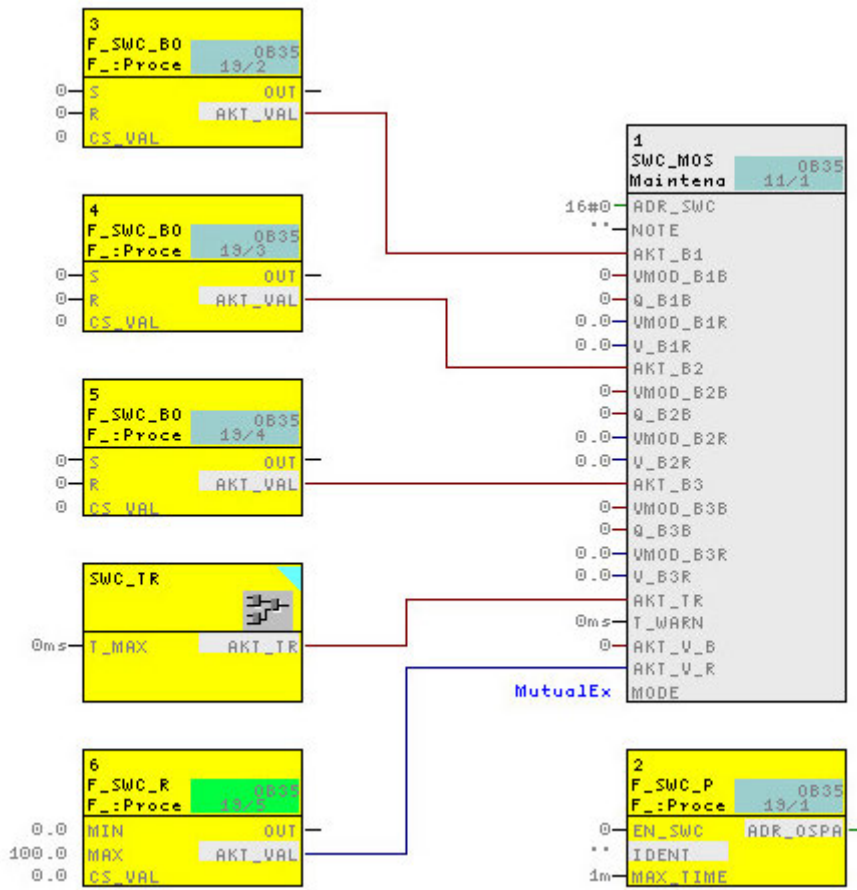
If you want to have the process value and its QUALITY displayed for the F-channel driver in the faceplate, connect the following outputs of the F-channel driver:

- Q_DATA or V_DATA output to the Q_BxB or V_BxR input of the SWC_MOS block.
- QUALITY output to the QUAL_Bx input of the SWC_MOS block

16. Compile your CFC chart.

Additional connections between the SWC_MOS block, the F-blocks F_SWC_BO or F_SWC_R, F_SWC_P and the F-channel drivers are created during compilation.

10.3 Programming operator functions




17. Follow the procedure as described in section "Configuring the faceplate of the operator functions (Page 140)".

10.3.2.7 Application case: Fail-safe acknowledgment

Application

This application case shows you how to perform a "Fail-safe acknowledgement" for a channel driver with the SWC_QOS block.

Core statement

| |
|--|
|  WARNING |
| Warnings in the descriptions of the F-blocks Observe the warnings in the descriptions of the F-block F_SWC_BO. |

1. Place the SWC_QOS block in your CFC chart.

Observe the information on assigning names in section "SWC_QOS: Operator function for fail-safe acknowledgment (Page 308)".

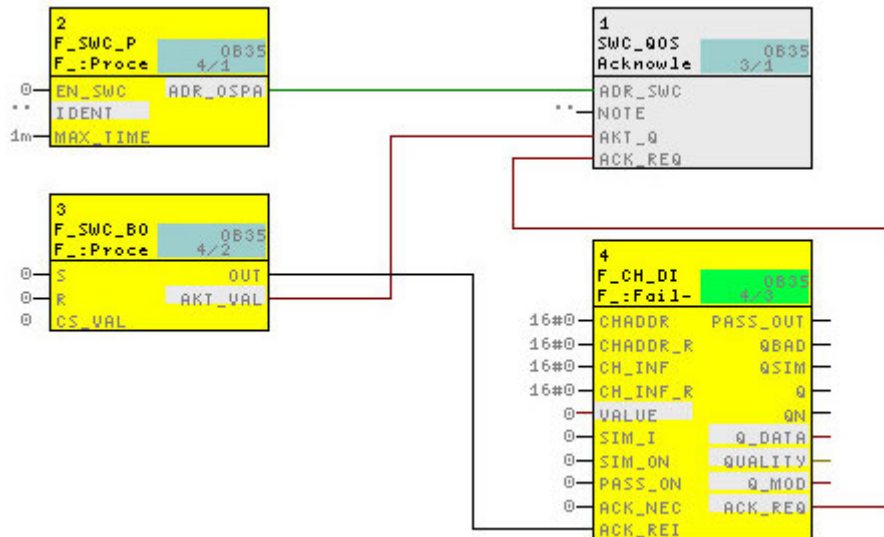
2. Place the F-block F_SWC_P, if necessary.

Optional:

If you place multiple protocol blocks F_SWC_P per shutdown group so that multiple simultaneous operator inputs are possible from the OS, connect the ADR_OSPA output of F_SWC_P to the ADR_SWC input of the associated operator control block SWC_QOS. This interconnection is not required when simultaneous operator inputs per shutdown group are not needed. You can find additional information on this in section "Introduction (Page 126)".

3. Place one F-block F_SWC_BO and F_CH_DI each.
4. Connect the EN_SWC input of the F-block F_SWC_P to a keyswitch.
5. Assign the maximum duration of the operator input (default setting is 1 minute) at the MAX_TIME input of the F-block F_SWC_P.
6. Connect the outputs of the F-block F_SWC_BO:
 - Connect OUT to the ACK_REI input of the F-block F_CH_DI
 - Connect AKT_VAL to the AKT_Q input of the SWC_QOS block
7. Connect the ACK_REQ output of F-block F_CH_DI to the ACK_REQ input of block SWC_QOS.
8. Before compiling, check the assignment of the SWC_QOS block. The block must be assigned to a standard runtime group.
9. Compile your CFC chart.

Additional connections between the SWC_QOS block, the F-blocks F_SWC_BO and F_CH_DI are created during compilation.



10. Follow the procedure as described in section "Configuring the faceplate of the operator functions (Page 140)".

10.3.3 Configuring the faceplate of the operator functions

A faceplate is created in the OS for each instance of an operator control block, e.g. SWC_MOS, in the safety program. The operator steps for "Secure Write Command++" are performed in the required sequence by one or two operators on the faceplate. The corresponding faceplate is called up in the OS via the associated block icon.

 **WARNING**

Restoration of edited faceplates

You can edit the faceplates of the operator functions.

If limitations arise, you can restore a backup copy of the respective file/function from the "Extras\FSYSTEMSHMI" directory of the product CD.

Requirements

- Placement, parameter assignment and interconnection of all required F-blocks, such as F_SWC_R, F_SWC_BO, in the CFC charts is complete.
For more information, refer to section "Placement, parameter assignment and interconnection of F-blocks in the CFC (Page 126)".
- The CFC charts with F-blocks for the desired operator function are located in the plant hierarchy.

Overview of configuring the faceplates in the ES

Configure the faceplates for the desired operator function, e.g. "Maintenance Override", on the ES with the following steps:

1. Creating block icons
2. Initializing properties of the block icons
3. Setting up authorizations for operators
4. Transferring configuration to the OS

The individual steps are described below.

Creating block icons

1. Open the PCS 7 project in SIMATIC Manager.
2. Create a new picture object in the level of the plant hierarchy containing the CFC charts with the F-blocks of the desired operator function.
3. Select the OS object in the project and select "Compile" from the shortcut menu to compile the OS.

Press the "Compile" button in the last dialog.

Result: When the OS is compiled, the block icons are automatically inserted in the new picture.

Initializing properties of the block icons

1. Double-click the picture file in the plant view of the PCS 7 project.

Result: WinCC Explorer is started and the picture file is displayed in the Graphics Designer. The name is displayed in the header of each block icon. The name of the block icon is formed from the name of the CFC chart and the name of the associated F-block instance.

2. Select a block icon and open the object properties.
3. Select the "Others" entry on the "Properties" tab.
4. Assign the desired authorizations to the "LevelInitiate", "LevelConfirm", "LevelBypass" and "LevelBypassValue" attributes.
 - The "LevelInitiate" and "LevelConfirm" attributes apply to the operator functions "Change process values", "Maintenance Override" and "Fail-safe acknowledgment".
 - The "LevelBypass" and "LevelBypassValue" attributes apply to the operator function "Maintenance Override".

Alternatively, you can accept the default authorizations for operators. See the next section "Setting up user authorizations for operators" for more information.

Default authorizations (correspond to the user hierarchies from PCS 7):

- For the operator that initiates a bypass or fail-safe value change with Maintenance Override (Initiator): No. 5, "Process controlling"
 - For the operator that initiates only a bypass with Maintenance Override (Bypass): No. 5, "Process controlling"
 - For the operator that initiates a fail-safe value change with Maintenance Override (BypassValue): No. 5, "Process controlling"
 - For the operator that confirms the bypass and a fail-safe value change with Maintenance Override (Confirmer): No. 6, "Higher process controlling"
5. Repeat Steps 2 and 4 for all block icons present.
 6. Save the picture file.

Setting up user authorizations for operators

An operator function is performed by two operators. For this purpose, create two users.

- The "Initiator" initiates the operator input, e.g. the bypass and/or the setting of bypass values in the case of "Maintenance Override".
- The "Confirmer" confirms this operator input.

Alternatively, the two steps can also be performed by only one operator. For this, create a user that has both "Initiator" and "Confirmer" authorizations.

Create the users with the following authorizations in WinCC Explorer using the "User Administrator" editor.

- For the "Maintenance Override" function

10.3 Programming operator functions

| User | Action | Required authorizations | | | |
|-----------------------|--|-------------------------|---------|--------|-------------|
| | | Initiate | Confirm | Bypass | BypassValue |
| Initiator | Set bypasses | X | — | X | — |
| | Set bypass values | X | — | — | X |
| | Set bypasses and bypass values | X | — | X | X |
| Confirmer | Confirm bypasses | — | X | X | — |
| | Confirm bypass values | — | X | — | X |
| | Confirm bypasses and bypass values | — | X | X | X |
| Initiator & Confirmer | Set and confirm bypasses | X | X | X | — |
| | Set and confirm bypass values | X | X | — | X |
| | Set and confirm bypasses and bypass values | X | X | X | X |

- For the "Change process values" function

| User | Action | Required authorizations | |
|-----------|----------------------|-------------------------|---------|
| | | Initiate | Confirm |
| Initiator | Change value | X | — |
| Confirmer | Confirm value change | — | X |

- For the "Fail-safe acknowledgment" function

| User | Action | Required authorizations | |
|-----------|----------------------|-------------------------|---------|
| | | Initiate | Confirm |
| Initiator | Change value | X | — |
| Confirmer | Confirm value change | — | X |

Activating the OS

Activate the WinCC Runtime system of the OS, e.g. by selecting **File > Activate** in WinCC Explorer.

Result

After activation, the hierarchy levels appear as buttons in the runtime system of the OS. Click the button to display the block icons for this level.

Example

The following figures show two block icons in the runtime system of the OS, dependent on the operator function.

Clicking a block icon opens the faceplate.

- Operator function "Change process values"

| SWC/SWC CHG R 01 | | | | SWC/SWC CHG R 01 | | | |
|------------------|--|----|--|------------------|--|----|--|
| CHG | | | | CHG | | | |
| 25 | | °C | | 35 | | °C | |

You can change F-parameters in the safety program. The successful change is visible from the changes in the last line.

- Operator function "Maintenance Override"

| MOS/SIM F-Kanal R | | | | MOS/SIM F-Kanal R | | | |
|-------------------|--|--|--|-------------------|--|--|--|
| MOS | | | | MOS | | | |

You can use "Maintenance Override" to establish a bypass of the F-channel drivers for maintenance work.

The following symbol in the block icon indicates an activated bypass.



- Operator function "Fail-safe acknowledgment"

| edgement/ACK_CH_DO | | | | edgement/ACK_CH_DO | | | |
|--------------------|--|--|--|--------------------|--|--|--|
| QOS | | | | QOS | | | |

The following symbol in the block icon indicates an acknowledgment request.



Detailed information

For detailed information on the described steps, refer to:

- "PCS 7 Operator Station (<https://support.industry.siemens.com/cs/ww/en/view/90682677>)" configuration manual
- Online help for the WinCC editors (e.g. Graphics Designer and User Administrator)

10.3.4 Integrating an operator function in an existing project

Introduction

You can also integrate an operator function such as "Maintenance Override" in an existing project.

Requirement

In order to integrate the operator function in an existing project, you must update your project.

Updating an existing project

1. Launch WinCC Explorer for the OS contained in the project.
2. Open the OS Project Editor.
3. Select the "Basic Data" tab.

If pictures from S7 F Systems HMI (picture "@PCS7Typicals_S7F_SDW.PDL" and all pictures "@PG_SWC_x.PDL") are already present in the project in the "Accept picture modules from the libraries" area, select these.

User-specific changes in these screens are lost.

4. Make sure that all other settings in the OS Project Editor conform to your specifications.
5. Then click the "OK" button.

The project is reconfigured and, as a result, the new block icons and the new pictures are applied.

6. Open the Global Script C Editor and select the menu command "Options > Regenerate Header".

Integrating an operator function

In order to introduce the new block icons into existing plant pictures, you must re-compile the relevant project.

1. Start SIMATIC Manager.
2. Select the OS object in the project and select the "Compile" menu item from the shortcut menu to compile the OS.
3. Click the "Compile" button in the last dialog of the "Compile OS" wizard.

Result

Once you have performed these steps, your project will contain the new block icons of the operator functions and the necessary pictures.

Note

If user settings for the block icon of an operator function are to be retained during a subsequent OS compilation of an existing picture, you must clear the "Derive block icons from the plant hierarchy" option for this WinCC picture.

10.4 Executing operator functions

10.4.1 Requirements and general notes

You carry out an operator input for a parameter in the OS by means of a faceplate. The operator input consists of a sequence of operations that can be performed by one or two operators.

Requirements

- The S7 program is compiled and downloaded to the F-CPU.
- The user(s) with the relevant authorizations are set up.
- The configuration of the faceplates is compiled and downloaded to the OS.
- When using OS clients, make sure that no default server is set for tags (in WinCC Explorer select "Server Data," in the shortcut menu select "Default Server" and in the "Configure Default Server" dialog for the "Tags" component select "No Default Server").

General information

WARNING

Initiator and confirmer must not accept an invalid value

Before starting the transaction, you must verify the following values in the faceplate:

- The technological name in the header of the faceplate.
- The name contained in the "ID" field (HID of the CPU or value of the "IDENT" parameter of the F_SWC_P).
- The "Tag name".

As the initiator or confirmer, you must not accept an invalid value. If there are inconsistencies, you must cancel the operation. As an operator, you must not rely on individual display fields of the faceplate; rather, you must check the values and compare them with each other.

WARNING

Technological assignment must be appropriate for the environment

When opening the faceplate, make sure that the technological assignment in the top line is appropriate for the environment in which the block icon was placed.



WARNING

Transaction for changing an F-Parameter

You can only perform one transaction for changing an F-Parameter at a time. You must use organizational measures to ensure that multiple transactions are not performed simultaneously for the same F-parameter. Otherwise, the transaction cannot be performed correctly, resulting in unexpected results, such as:

- Display of incorrect values in the faceplate fields
- Or
- Unexpected cancellation of the transaction

If an operation is already active

If an operation for another faceplate is already in progress, the message "Other command function active" appears when opening the faceplate in WinCC Runtime.

Note

The message "Other operator function active" appears when two operator control blocks are assigned to the same protocol block and both want to perform an operator input at the same time.

Starting from S7 F Systems V6.2, multiple operator functions can be executed simultaneously. You can find additional information in section "Introduction (Page 126)" in paragraph "Multiple protocol blocks in a shutdown group".

10.4.2 Use of operator function "Change process value" with two operators

Operator authorizations

To change a process value, two operators having different authorizations are required.

- The Initiator initiates the process value change. The authorization corresponds to the "InitiatorAuthorization" attribute in the properties of the block icon. The default setting is No. 5, "Process controlling".
- The Confirmer verifies and confirms the change. This operator must have the necessary authorization for confirming the change but not for initiating it. The authorization corresponds to the "ConfirmerAuthorization" attribute in the properties for the block icon. The default setting is No. 6, "Higher process controlling".

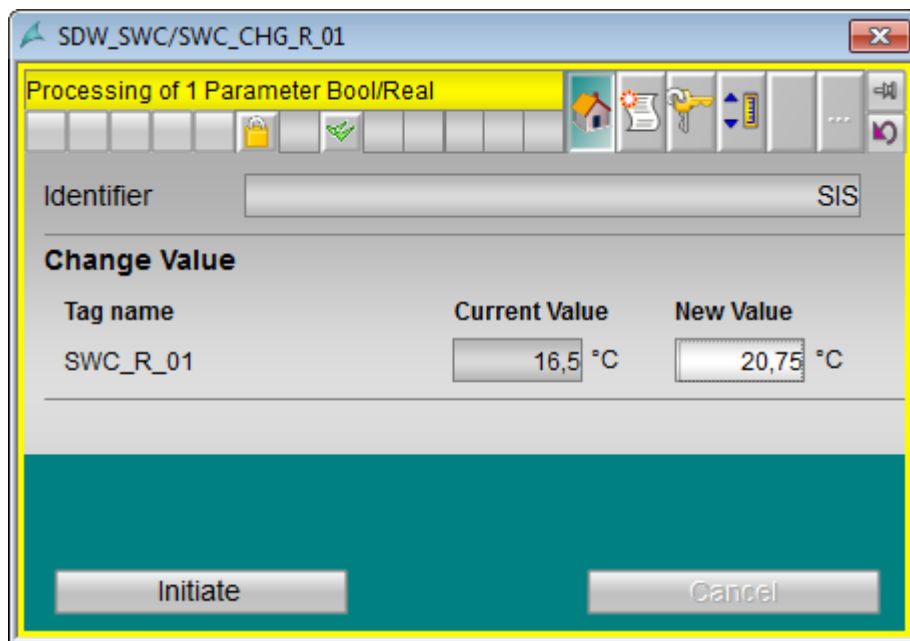
Note

The sections below describe the necessary operator input steps for the two operators. The figures show the example of an F_REAL parameter with the operator identifiers (Login):

- level5 – Initiator
- level6 – Confirmer

Initiator: Initiate value change

1. Log on to the OS as a user with "Initiator" authorization.
2. Click the desired block icon to open the faceplate.



You can check the authorization of the logged on operator in the "User rights" view of the faceplate, which is opened by the button of the same name in the toolbar.

3. To perform a value change, enter the desired value in the "New value" text box and confirm the input by pressing the <Enter> key. If you are changing an F_REAL value, the configured "MIN", "MAX" and "MAXDELTA" values are evaluated.

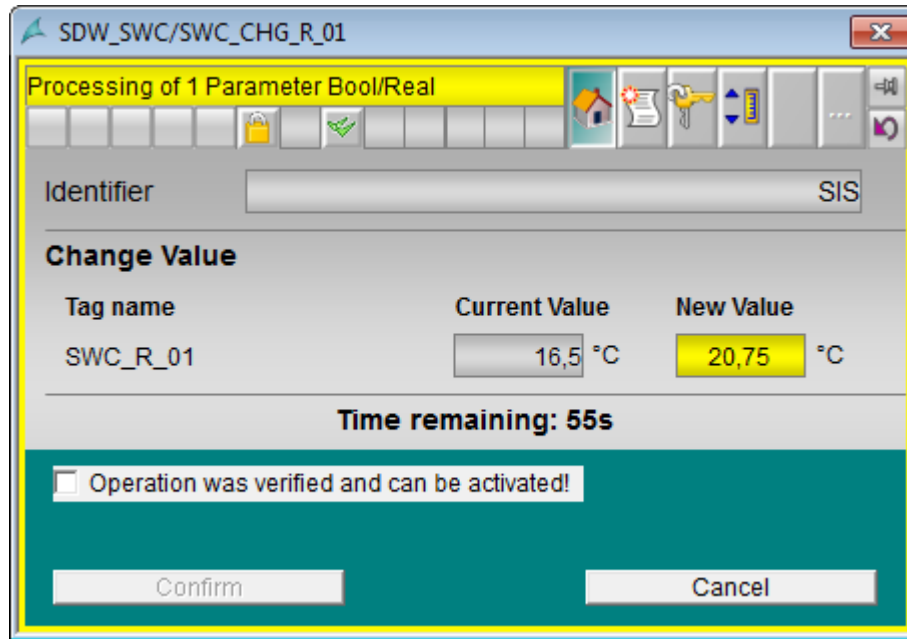
You can check the current limits in the "Limits" view of the faceplate, which is opened by the button of the same name in the toolbar.

4. Click the "Initiate" button.

The Confirmer must then continue the operator input. If you cancel the operator input after pressing the "Initiate" key, check whether the previously valid value is displayed in the "Current value" field.

Confirmer: Confirm value change

1. Log on to the OS as a user with "Confirmer" authorization.
You can log on to a second OS or on the same OS as the Initiator.
2. Click the desired block icon to open the faceplate.

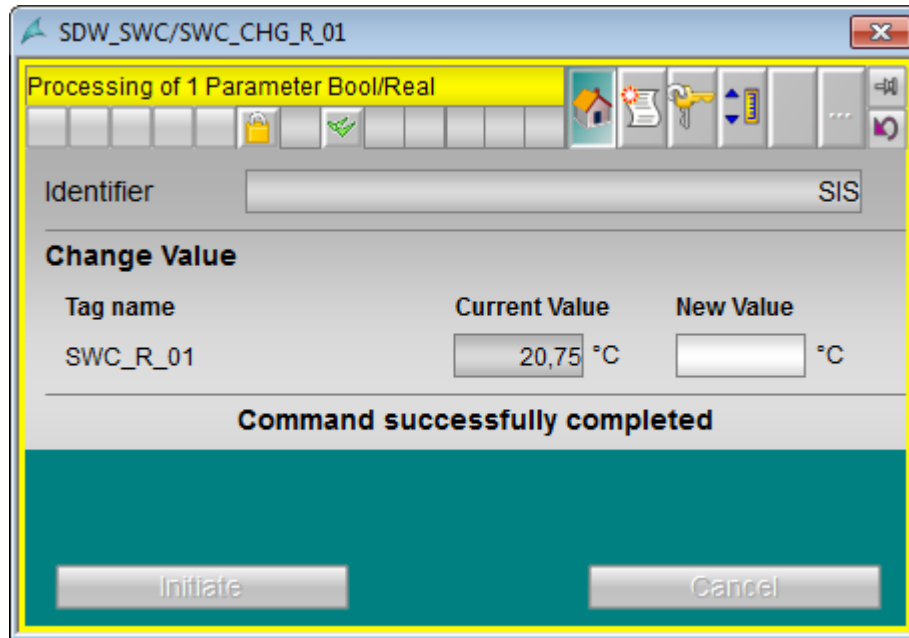


You can check the authorization of the logged on operator in the "User rights" view of the faceplate.

3. Verify that:
 - The right operator control block was selected (technological name in the header of the faceplate).
 - The right F-CPU was selected (for identifier, refer to section "F_SWC_P: Centralized control of operator input via the OS (Page 282)").
 - The right parameter is to be changed (tag name).
 - The change (modified value) is displayed correctly.
 - New values of the changed parameters are highlighted in yellow under "New value".
4. Confirm the change with "Operation was verified and can be activated!" or cancel the operator input with the "Cancel" button.
5. Press the "Confirm" button to activate the value change. Click "Cancel" to cancel the operation.

Result

The successful value change is signaled. The entry in the "New value" text box has been applied to the field under "Current value".



10.4.3 Use of operator function "Change process value" with one operator

Authorization for the operator

If the "Change process value" operator input is performed by only one operator, this operator must have the "Initiator" and "Confirmer" authorizations.

For this purpose, create an operator assigned the "LevelInitiate" and "LevelConfirm" levels in the properties of the block icon. Further information on this can be found in section "Configuring the faceplate of the operator functions (Page 140)".

Changing process values with only one operator

The sequence is the same as when the operation is carried out with two operators, but one operator can perform all the steps (see also section "Use of operator function "Change process value" with two operators (Page 146)").

The difference is that there is no longer a wait for the Confirmer. Instead, the operator can verify and confirm the operator input immediately after pressing the "Initiate" button.

All other steps remain the same.

10.4.4 Use of operator function "Maintenance Override" with two operators

Operator authorizations

The "Maintenance Override" operator function allows you to set bypasses in the safety program from the OS.

Two operators having different authorizations are required to create a bypass.



- The Initiator initiates the bypass of the F-channel driver. This operator must have the "LevelInitiate", "LevelBypass" and "LevelBypassValue" authorizations for initiating the bypass but not for confirming it.
- The Confirmer verifies and confirms the change. This operator must have the required "LevelConfirm", "LevelBypass" and "LevelBypassValue" authorizations for confirming the change but not for initiating it.

Reset time

If you have configured a retrigger function in the CFC chart, the simulation is only active for the time configured at the T_MAX input of the chart-in-chart block SWC_TR. As the Initiator, if you click the "Retrigger" button while the configured reset time is running, the reset time restarts with the configured time after the change is confirmed by the Confirmer.

Quality of the process value on the F-Channel driver

The quality of the process value on the F-Channel driver is indicated in the faceplate by the following symbols:

| Symbol | State | Quality code |
|---|------------------------|--------------|
| No symbol | Valid value | 16#80 |
|  | Simulation | 16#60 |
|  | SUBSTITUTION VALUE | 16#48 |
| | Last valid value | 16#44 |
| | Invalid value (F-STOP) | 16#00 |

See also section "F-Channel drivers for F-I/O (Page 310)".

Value on the F-Channel driver

If the V_MOD_Bx inputs are interconnected on the SWC_MOS block, the values on the F-Channel drivers are displayed under V_MOD.

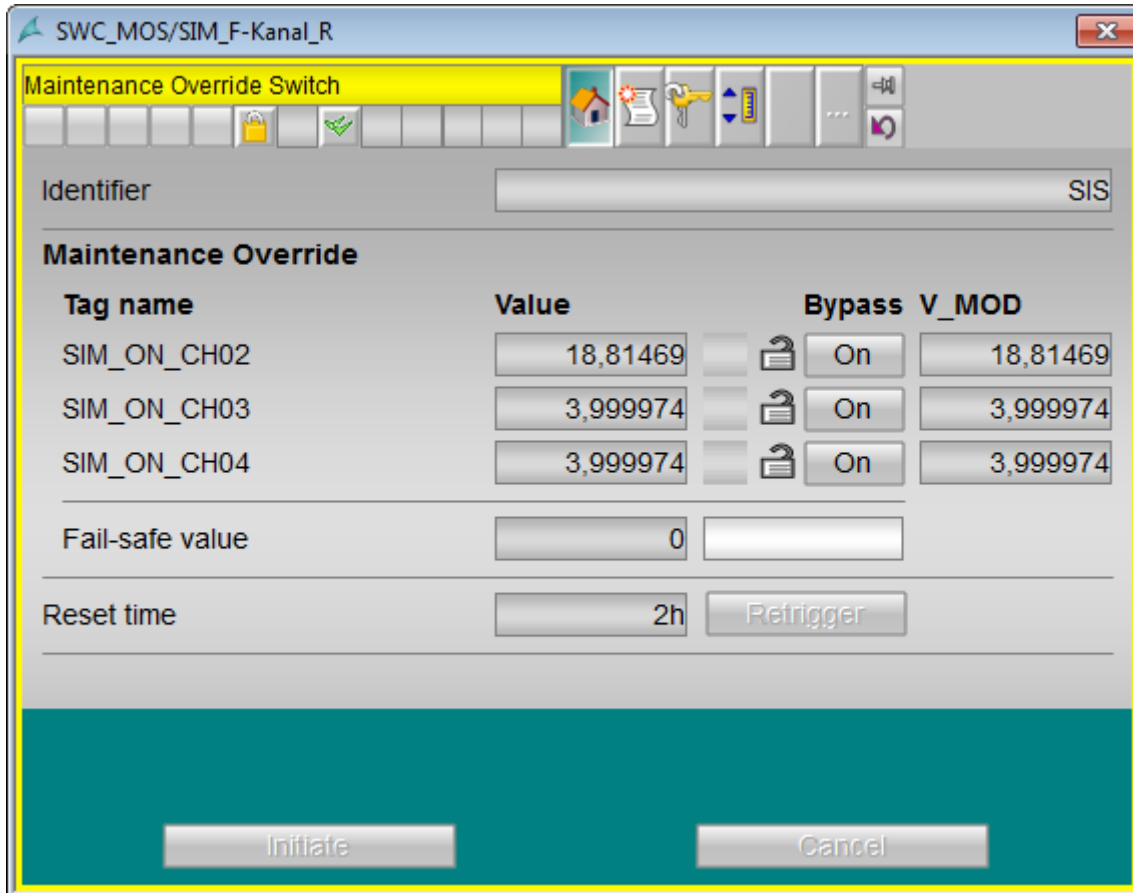
Note

The sections below describe the necessary transaction steps for the two operators. The figures show the example of an F_REAL parameter with the operator identifiers (Login):

- level5 – Initiator
 - level6 – Confirmer
-

Initiator: Initiating a bypass

1. Log on to the OS as a user with "Initiator" authorization.
2. Click the desired block icon to open the faceplate.




You can check the authorization of the logged on operator in the "User rights" view of the faceplate, which is opened by the button of the same name in the toolbar.

Under "Value" on the Maintenance Override faceplate, you can see the current process value of the F-I/O and the current fail-safe value setting. The values on the F-Channel drivers are displayed in the V_MOD column.

The symbols under "Bypass" show you the current status of the bypass (SIM_ON) on the F-channel drivers:

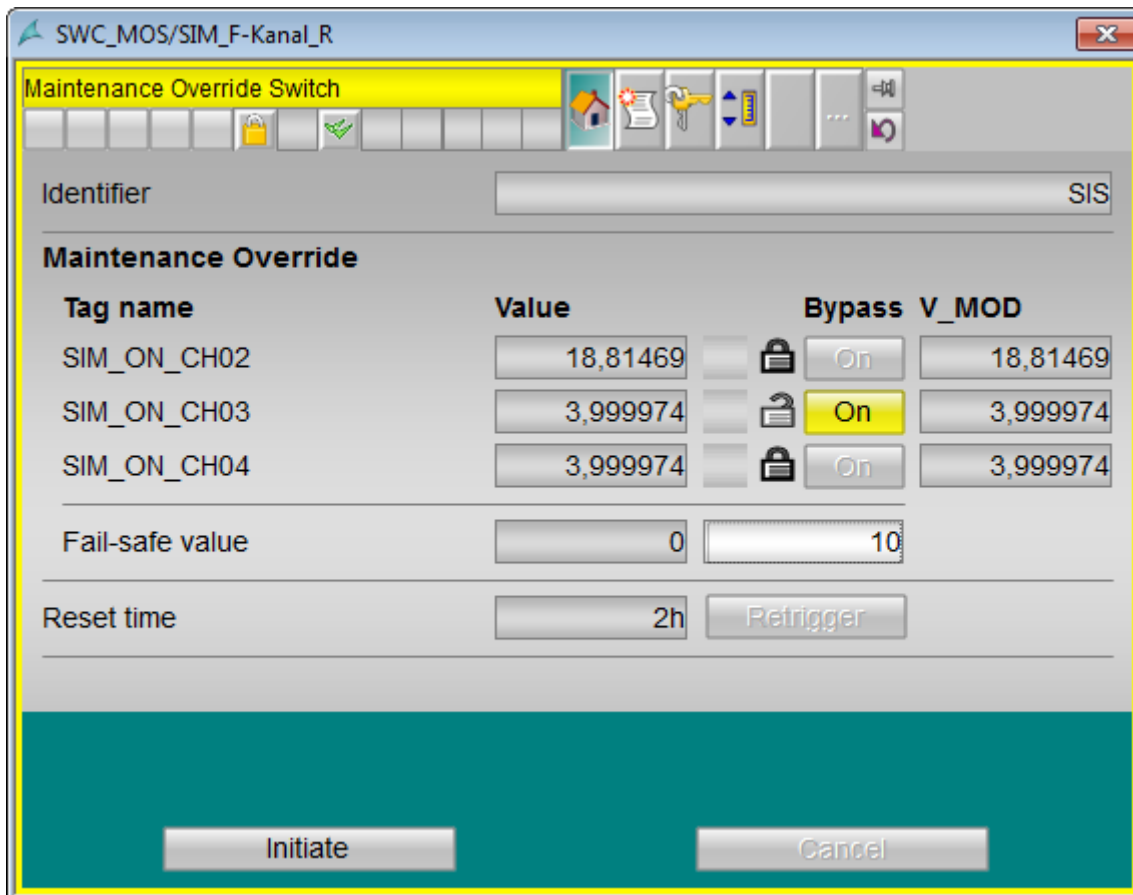
| Symbol | Meaning |
|--------|--|
| | Bypass not active |
| | Bypass active |
| | A bypass can be created for this F-Channel driver. |
| | For this F-Channel driver, either a bypass cannot be created (mutually exclusive interlock) or the user authorization is insufficient. |

3. To enable a bypass for one or more F-channel drivers, press the corresponding button under "Bypass".
4. If the input setting MODE = 'MutualExclBypass' has been assigned on the SWC_MOS block, the remaining F-Channel drivers are interlocked when a bypass is enabled. The interlocked F-channel drivers are indicated by the following symbol.

5. If you want to change the current fail-safe value on F-channel drivers for F_BOOL, press the button under "Bypass".

If you are using F-channel drivers for F_REAL and want to change the fail-safe value, enter the new fail-safe value in the text box and confirm your input with the <Enter> key. The configured "MIN" and "MAX" limits are evaluated in the process.

You can check the current limits in the "Limits" view of the faceplate, which is opened by the button of the same name in the toolbar.

6. If you want to reset the reset time to the configured initial value, click the "Retrigger" button.



SWC_MOS/SIM_F-Kanal_R

Maintenance Override Switch

Identifier: SIS

| Tag name | Value | Bypass | V_MOD |
|-------------|----------|--------|----------|
| SIM_ON_CH02 | 18,81469 | On | 18,81469 |
| SIM_ON_CH03 | 3,999974 | On | 3,999974 |
| SIM_ON_CH04 | 3,999974 | On | 3,999974 |

Fail-safe value: 0 10

Reset time: 2h Retrigger

Initiate Cancel

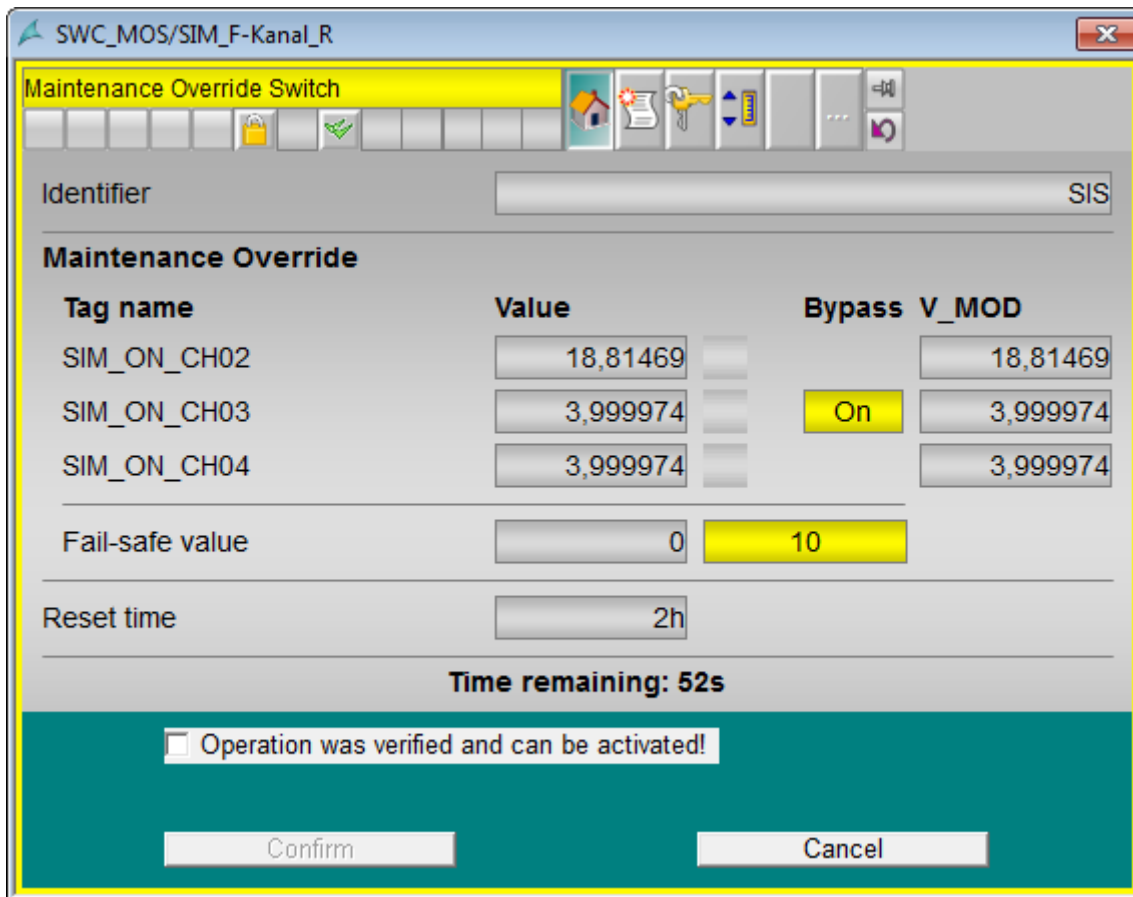
7. Click the "Initiate" button.

The Confirmer must then continue the operator input.

If you cancel the operator input after pressing the "Initiate" button, check whether the previously valid value is displayed in the "Value" field.

Confirmer: Confirming a bypass

1. Log on to the OS as a user with "Confirmer" authorization.
You can log on to a second OS or on the same OS as the Initiator.
2. Click the desired block icon to open the faceplate.



You can check the authorization of the logged on operator in the "User rights" view of the faceplate.

3. Verify that:
 - The right operator control block was selected (technological name in the header of the faceplate).
 - The right F-CPU was selected (for identifier, refer to section "F_SWC_P: Centralized control of operator input via the OS (Page 282)").
 - The right parameter is to be changed (tag name).
 - The change (modified value) is displayed correctly.
 - New values of the changed parameters are highlighted in yellow under "Bypass".
 - No other fields for new values are highlighted in yellow.

4. Confirm the change with "Operation was verified and can be activated!" or cancel the operator input with the "Cancel" button.
5. Click "Confirm" to enable the bypass. Click "Cancel" to cancel the operation.

Result

The successful change on the F-Channel drivers is signaled. The F-channel driver for which the bypass was activated is indicated with the following symbol.



Depending on the interconnection on SWC_MOS, additional status displays become visible (see section "SWC_MOS: Command function for Maintenance Override (Page 307)").

If you have configured a reset time, the countdown for this time begins. Bypasses are automatically canceled when the reset time has elapsed.

SWC_MOS/SIM_F-Kanal_R

Maintenance Override Switch

Identifier: SIS

Maintenance Override

| Tag name | Value | Bypass | V_MOD |
|-------------|----------|--------|----------|
| SIM_ON_CH02 | 18,81469 | On | 18,81469 |
| SIM_ON_CH03 | 10 | Off | 3,999974 |
| SIM_ON_CH04 | 3,999974 | On | 3,999974 |

Fail-safe value: 10

Reset time: 1h59m52s [Retrigger]

Command successfully completed

[Initiate] [Cancel]

10.4.5 Use of operator function "Maintenance Override" with one operator

Authorization for the operator

The "Maintenance Override" operator function allows you to set bypasses in the safety program from the OS.

If the bypass of the F-channel driver is implemented by only one operator, this operator must be authorized to both initiate and confirm the bypass.

For this purpose, create an operator assigned the "LevelInitiate", "LevelConfirm", "LevelBypass" and "LevelBypassValue" levels in the properties of the block icon. Further information on this can be found in section "Configuring the faceplate of the operator functions (Page 140)".

Creating a bypass with only one operator

The sequence is the same as when the operation is carried out with two operators, but one operator can perform all the steps (see also section "Use of operator function "Maintenance Override" with two operators (Page 150)").

The difference is that there is no longer a wait for the Confirmer. Instead, the operator can verify and confirm the operator input immediately after pressing the "Initiate" button.

All other steps remain the same.

10.4.6 Use of operator function "Fail-safe acknowledgment" with two operators

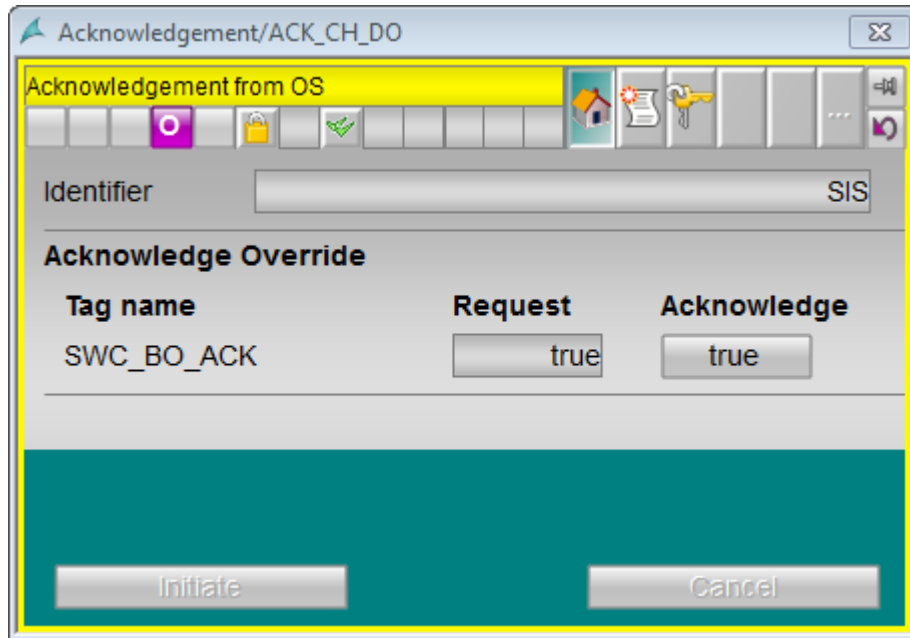
Operator authorizations

The fail-safe acknowledgment requires two operators having different authorizations.

- The Initiator initiates the fail-safe acknowledgment. The authorization corresponds to the "InitiatorAuthorization" attribute in the properties of the block icon. The default setting is No. 5, "Process controlling".
- The Confirmer verifies and confirms the acknowledgment. This operator must have the necessary authorization for confirming the change but not for initiating it. The authorization corresponds to the "ConfirmerAuthorization" attribute in the properties for the block icon. The default setting is No. 6, "Higher process controlling".

Initiator: Initiating a fail-safe acknowledgment

1. Log on to the OS as a user with "Initiator" authorization.
2. Click the desired block icon to open the faceplate.



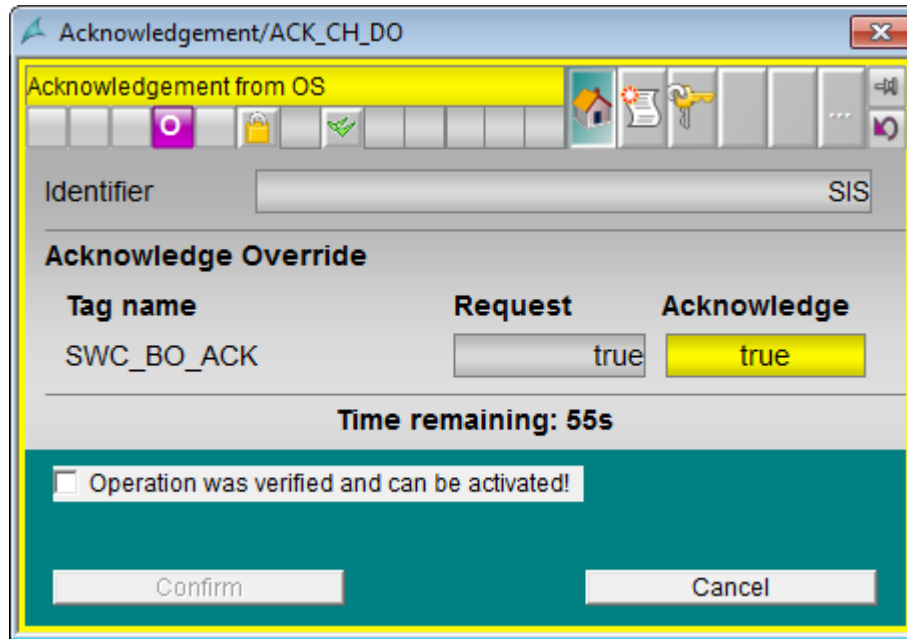
You can check the authorization of the logged on operator in the "User rights" view of the faceplate, which is opened by the button of the same name in the toolbar.

3. To perform a fail-safe acknowledgment, click the button below "Acknowledge". The button background turns yellow.
4. Click the "Initiate" button.

The Confirmer must then continue the acknowledgment.

Confirmer: Confirming the fail-safe acknowledgment

1. Log on to the OS as a user with "Confirmer" authorization.
You can log on to a second OS or on the same OS as the Initiator.
2. Click the desired block icon to open the faceplate.

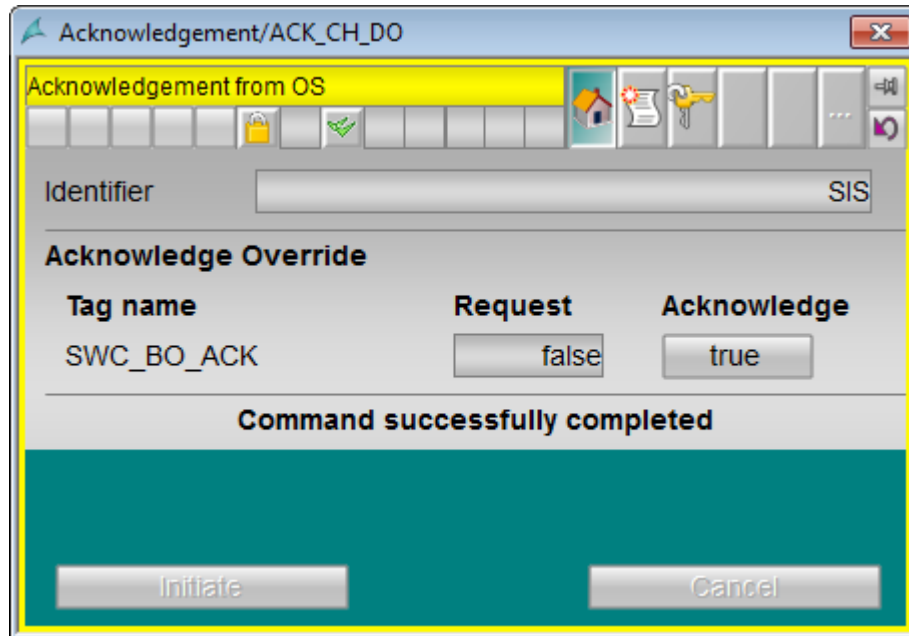


You can check the authorization of the logged on operator in the "User rights" view of the faceplate.

3. Verify that:
 - The right operator control block was selected (technological name in the header of the faceplate).
 - The right F-CPU was selected (for identifier, refer to section "F_SWC_P: Centralized control of operator input via the OS (Page 282)").
 - The right parameter is to be changed (tag name).
 - The new value of the changed parameter is highlighted in yellow under "Acknowledge".
4. Confirm the change with "Operation was verified and can be activated!" or cancel the operator input with the "Cancel" button.
5. Press the "Confirm" button to confirm the fail-safe acknowledgment. Click "Cancel" to cancel the operation.

Result

The successful operator input is signaled.



See also

Configuring the faceplate of the operator functions (Page 140)

10.4.7 Use of operator function "Fail-safe acknowledgment" with one operator

Authorization for the operator

If the fail-safe acknowledgment is performed by only one operator, this operator must have the "Initiator" and "Confirmer" authorizations.

For this purpose, create an operator assigned the "LevelInitiate" and "LevelConfirm" levels in the properties of the block icon. Further information on this can be found in section "Configuring the faceplate of the operator functions (Page 140)".

Fail-safe acknowledgment with only one operator

The sequence is the same as when the operation is carried out with two operators, but one operator can perform all the steps (see also section "Use of operator function "Fail-safe acknowledgment" with two operators (Page 156)").

The difference is that there is no longer a wait for the Confirmer. Instead, the operator can verify and confirm the operator input immediately after pressing the "Initiate" button.

All other steps remain the same.

Safety Data Write function: Changing F-parameters from the OS

11

11.1 Safety Data Write concept

Function

The "Safety Data Write" functionality enables safety-related changes to be made to F-parameters in the safety program of an F-CPU from an operator station (OS).

A special safety protocol is used for changing F-Parameters during safety mode operation. This ensures compliance with the safety requirements of Safety Integrity Level up to SIL3 in accordance with IEC 61508:2010. The modified F-Parameter values can be retained even after a warm restart of S7 F/FH systems.

Note

Availability and use

- Availability:
The "Safety Data Write" functionality is available only under SIMATIC PCS 7.
 - Use:
In place of the "Safety Data Write" functionality, starting from *S7 F Systems V6.2* and *S7 F Systems Lib V1_3 SP2*, it is possible to configure and execute safety-related changing of F-parameters in the safety program using the "Change process values" operator function of "Secure Write Command++".
Further information can be found in section "Concept of "Secure Write Command++" (Page 123)".
-

The *S7 F Systems* optional software offers the following for Safety Data Write:

- Two F-blocks that you must integrate in the CFC charts of your safety program
 - F_CHG_R: Safety Data Write for F-Parameters of data type F_REAL
 - F_CHG_BO: Safety Data Write for F-Parameters of data type F_BOOL
- The associated faceplates that you must integrate in your OS

Transaction for Safety Data Write

Safety Data Write allows an F-Parameter in the safety program of an F-CPU to be changed, provided a certain operating sequence is carried out in the OS within a certain time. The entire change operation is referred to as a "transaction".

Operator Types for Safety Data Write

A transaction can be performed by an individual operator who initiates, verifies, and confirms the change. However, a transaction can also be performed by two operators. One operator (the initiator) initiates the change, and the second (the confirmer) re-enters, verifies, and confirms this value.

11.2 Programming Safety Data Write

11.2.1 Basic procedure

Basic procedure

To implement Safety Data Write by means of an OS, you must perform the following steps:

On the ES

1. Insert the F-Blocks F_CHG_R and F_CHG_BO into the *CFC chart* and interconnect them.
2. Configure the faceplate for Safety Data Write.

On the operator station (OS)

- Change the F-Parameters with Safety Data Write.

The individual steps are described in detail in the sections below.

11.2.2 Positioning, interconnecting, and assigning parameters to F-Blocks in the CFC chart

Application

You can make changes to F-Parameters of the safety program by means of Safety Data Write using the F-Blocks F_CHG_R and F_CHG_BO.

Procedure

WARNING

Warnings in the descriptions of F-Blocks

Make sure that you comply with the warnings in the descriptions of the F_CHG_R and F_CHG_BO F-Blocks.

1. Insert one F_CHG_R or F_CHG_BO F-Block, respectively, for each input of data type F_REAL or F_BOOL that is to be changed using Safety Data Write (see Example 1: F_CHG_R (Page 165) and Example 2: F_CHG_BO (Page 165)).
2. Interconnect the OUT output to the input whose value you want to change using Safety Data Write.
3. Assign a pair of numbers to the SAFE_ID1 and SAFE_ID2 inputs. This ensures the association between the instance of F_CHG_R/F_CHG_BO and the corresponding faceplate. SAFE_ID1 must be unique from all others in the program. The pair of numbers

for SAFE_ID1 and SAFE_ID2 must be unique from all others in the system. You must configure the same pair of numbers on the block icon of the associated faceplate.

4. Interconnect the EN_CHG input to the enable signal for Safety Data Write.
5. Assign the maximum permissible time for the duration of the transaction to the TIMEOUT input. The transaction starts as soon as the initiator has accepted his entry.

All steps for verifying the transaction must be taken into account when configuring this time. For example, if two operators are required to enable the change, an appropriate amount of time must be allotted for both operators to log on and perform the necessary steps.

6. For F_CHG_R only: Assign limit values to the MIN and MAX inputs to specify the time during which the F-Parameters (output OUT) can be changed.
7. For F_CHG_R only: Assign the value of the maximum permissible increment of the change to the MAXDELTA input to specify the amount by which the F-Parameter (output OUT) can change relative to the current existing value.
8. Assign the initial value to input CS_VAL that is to be applied to output OUT in the event of a cold restart.
For F_CHG_R only: When a cold restart occurs, CS_VAL is applied at output OUT irrespective of the values for MIN and MAX. The configured value at input CS_VAL must be between the MIN and MAX values.
9. Optional: Assign 0 to input WS_MODE if the value at input CS_VAL is also to be applied to output OUT during a warm restart. The default value of input WS_MODE is 1.
10. Optional: Evaluate the CS_USED output in the safety program if you need to respond differently after an F-Startup in your safety program depending on whether the CS_VAL value or the last valid value at the OUT output has been made available.
11. For F_CHG_R only: Set the unit of measurement for the F-Parameter to be changed.

To do so, open the properties for the F-Block and select output CURR_R in the "Outputs" tab. In the "Unit" field, select the desired unit of measurement (e.g., kg/min) from the drop-down list.

The unit is displayed on the faceplate in the OS.

11.2.3 Examples: Safety Data Write

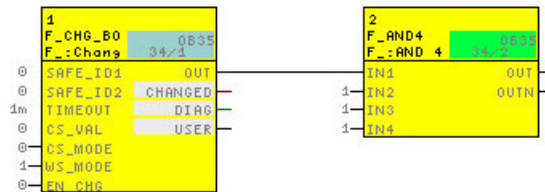
11.2.3.1 Example 1: F_CHG_R

The following figure shows an instance of F_CHG_R. The OUT output is interconnected to the "U_HL" input of F_LIM_HL whose value is to be changed in a fail-safe manner using Safety Data Write.



11.2.3.2 Example 2: F_CHG_BO

The following figure shows an instance of F_CHG_BO. The OUT output is interconnected to the "IN1" input of F_AND4 whose value is to be changed in a fail-safe manner using Safety Data Write.



11.2.4 Configuring the Faceplate for Safety Data Write.

A faceplate must be created in the OS for each instance of an F-block F_CHG_R and F_CHG_BO in the safety program. The operator steps for the Safety Data Write transaction are performed on the faceplate in the required sequence by one or two operators. The corresponding faceplate is called up in the OS via the associated block icon.

Requirements

- Placement, parameter assignment and interconnection of all required F-blocks F_CHG_R and F_CHG_BO in the CFC charts is complete.
- The CFC charts with the F_CHG_R and F_CHG_BO F-blocks are located in the plant hierarchy.
- The safety program is compiled.

Configuring faceplates in the ES

Configure the faceplates for Safety Data Write on the ES with the following steps:

1. Creating block icons
2. Initializing properties of the block icons
3. Setting up authorizations for operators
4. Transferring configuration to the OS

The individual steps are described below.

Creating block icons

1. Open the *PCS 7* project in *SIMATIC Manager*.
2. Create a new picture object in the level of the plant hierarchy containing the CFC charts with the F-blocks F_CHG_R and F_CHG_BO.
3. Select the picture object and open the object properties.
4. In the "Block Icons" tab, activate the "Derive block icons from the plant hierarchy" option.
5. Click "OK" or "Apply" to confirm the revised properties.
6. Select the OS object and select "Compile" from the shortcut menu to compile the OS.
7. If necessary, select the "Create/update block icons" option in the "Compile OS" wizard when selecting the data you want to compile and the scope of the compilation. Press the "Compile" button in the last dialog.

Result: When the OS is compiled, the block icons are automatically inserted in the new picture.

Note


To prevent overwriting SAFE_ID1 and SAFE_ID2, deactivate the "Derive block icons from the plant hierarchy" option in the object properties for the WinCC picture before recompiling the OS.

Initializing properties of the block icons

1. Double-click the picture file in the plant view of the *PCS 7* project.

Result: WinCC Explorer is started and the picture file is displayed in the Graphics Designer. The name is displayed in the header of each block icon. The name of the block icon is formed from the name of the CFC chart and the name of the associated F-block instance.

2. Select a block icon and open the object properties.
3. Select "User configuration" on the "Properties" tab.
4. Assign the exact static values to the SAFE_ID1 and SAFE_ID2 attributes that are configured for the SAFE_ID1 and SAFE_ID2 inputs of the associated F-block instance.

| |
|--|
|  WARNING |
| <p>Static values of the SAFE_ID1 and SAFE_ID2 attributes</p> <p>The static values of the SAFE_ID1 and SAFE_ID2 attributes must be identical to the F-parameters that are configured for the SAFE_ID1 and SAFE_ID2 inputs of the associated F-block instance.</p> <p>Note that you must enter these values for the F-blocks in the <i>CFC Editor</i> and for the block icons in <i>WinCC</i> independently and separately.</p> |

- Assign the desired authorizations to the "InitiatorAuthorization" and "ConfirmerAuthorization" attributes. Alternatively, you can accept the default authorizations for operators. See also "Setting up authorizations for operators".

Default authorizations (correspond to the user hierarchies from *PCS 7*):

- For the operator who initiates the change to an F-parameter using Safety Data Write (Initiator): No. 5, Process controlling
- For the operator who confirms the change to an F-parameter using Safety Data Write (Confirmer): No. 6, Higher process controlling

- Repeat Steps 2 and 5 for all block icons present.
- Save the picture file.

Examples

| | |
|------------------|------------------|
| SDW_1/SDW_F_BOOL | SDW_1/SDW_F_REAL |
| 0 | 0.000 |
| SAFE_ID1: 2 | SAFE ID1: 1 |
| SAFE_ID2: 3 | SAFE ID2: 0 |

Figure 11-1 Example: Block icons in a picture file

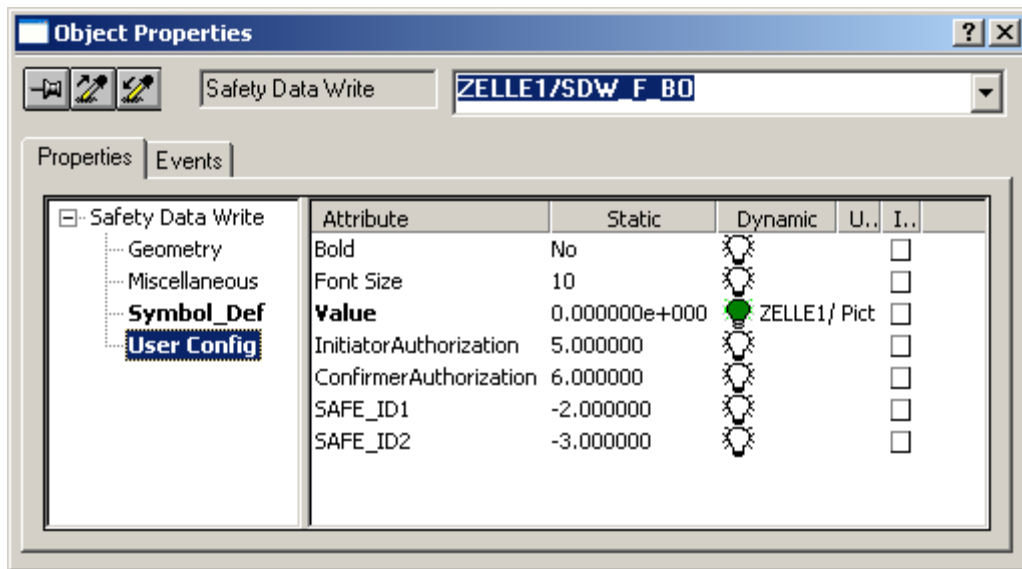


Figure 11-2 Example: Properties of a block icon

Setting up user authorizations for operators

Create the following users based on whether the transaction is to be performed by two operators or by one operator only:

- If the transaction for an F-parameter is to be performed by two operators, create two users:
 - The Initiator initiates the change to an F-parameter using Safety Data Write. This user must have the authorization assigned to the "InitiatorAuthorization" attribute in the properties for the block icon. However, the Initiator is not authorized to confirm the change.
 - The Confirmer verifies and confirms the change. This user must have the authorization assigned to the "ConfirmerAuthorization" attribute in the properties for the block icon. However, the "Confirmer" is not authorized to initiate the change.
- If only one operator is to perform all of the transaction steps, create a user who has both authorizations assigned to the "InitiatorAuthorization" and "ConfirmerAuthorization" attributes in the properties for the block icon.

Create the users in WinCC Explorer using the "User Administrator" editor.

Activating the OS

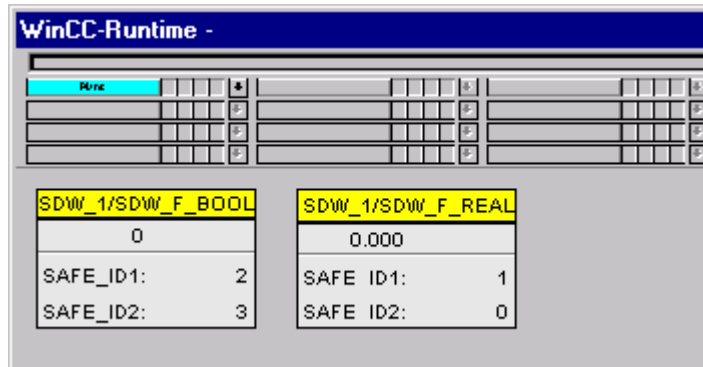
Activate the WinCC Runtime system of the OS, e.g. by selecting **File > Activate** in WinCC Explorer.

Result

After activation and login, the hierarchy levels appear as buttons in the runtime system of the OS. Click the button to display the block icons for this level.

Example

The following figure shows two block icons in the runtime system of the OS.



Clicking a block icon opens the faceplate that you can use to change an F-parameter by means of Safety Data Write.

Detailed information

For detailed information on the described steps, refer to:

- "PCS 7 Operator Station (<https://support.industry.siemens.com/cs/ww/en/view/90682677>)" configuration manual
- Online help for the WinCC editors (e.g. Graphics Designer and User Administrator)

11.3 Changing F-Parameters with Safety Data Write

11.3.1 Requirements and General Instructions

You perform a transaction for changing an F-Parameter using Safety Data Write by means of a faceplate in the OS. The transaction consists of a sequence of operations that can be performed by one or two operators.

Requirements

- The S7 program is compiled and downloaded to the F-CPU.
- The user(s) with the relevant authorizations are set up.
- The configuration of the faceplates is downloaded to the OS.
- The AS/OS connection is okay. The operator can test the AS/OS connection using the "OS Test" button (see the section entitled "Testing AS/OS connection" below).
- The EN_CHG input of the F-Block instance of F_CHG_R or F_CHG_BO for enabling Safety Data Write is set to TRUE.
- When using OS clients, make sure that no default server is set for tags (in WinCC Explorer select "Server Data," in the shortcut menu select "Default Server" and in the "Configure Default Server" dialog for the "Tags" component select "No Default Server").

Specifications for Changing an F-Parameter using Safety Data Write

The operator(s) need the following information to change an F-Parameter using Safety Data Write:

- Name of the block icon
- New value for the F-Parameter

General Information

The transaction must be completed within a specified time interval (Timeout). If the transaction is not finished before the Timeout interval elapses, the transaction is automatically canceled once the Timeout interval expires.

WARNING

Initiator and confirmer must not accept an invalid value

As the initiator or confirmer, you must not accept an invalid value. If there are inconsistencies, you must cancel the transaction.

As an operator, you must not rely on individual display fields of the faceplate; rather, you must check the values and compare them among each other.

Before starting the transaction, you must verify the plant name in the header of the faceplate.

WARNING

Technological assignment must be appropriate for the environment

When opening the faceplate, make sure that the technological assignment in the top line is appropriate for the environment in which the faceplate was placed.

WARNING

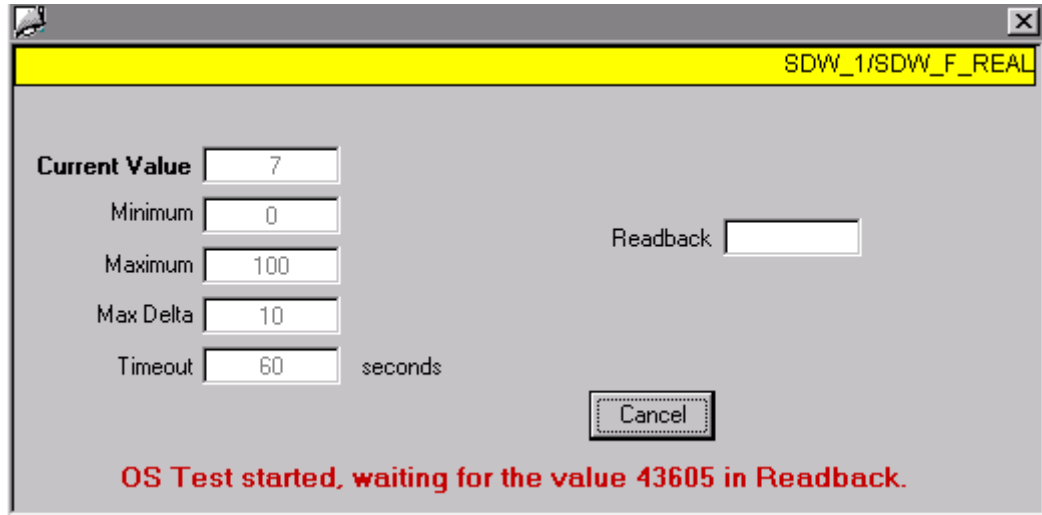
Transaction for changing an F-Parameter

You can only perform one transaction for changing an F-Parameter at a time. You must use organizational measures to ensure that multiple transactions are not performed simultaneously for the same F-Parameter. Otherwise, the transaction cannot be performed correctly, resulting in unexpected results, such as:

- Display of incorrect values in the faceplate fields
- or*
- Unexpected cancellation of the transaction

Testing AS/OS Connection

Before starting the transaction, you can test the AS/OS connection by clicking the "OS Test" button.



If the AS/OS connection is okay, a message to that effect is output and the expected value is displayed in the "Read Back" field.

If the AS/OS connection is not okay, the following error message is displayed: "OS test failed".

If the block is assigned

If a transaction for a faceplate has been started already, the following message appears when opening the faceplate in WinCC Runtime:

"Block is assigned. Please wait..."

To start a new transaction, click "Cancel" and reopen the faceplate.

11.3.2 Changing an F-Parameter with Two Operators

Operator authorizations

The transaction requires two operators having different authorizations.

- The initiator initiates a change to an F-Parameter using Safety Data Write. This user must have the authorization for initiating the change but not for confirming it. The authorization corresponds to the "InitiatorAuthorization" attribute in the properties for the block icon. The default setting is no. 5, process controlling.
- The confirmer enters the modified value again, verifies it, and confirms the change. This user must have the necessary authorization for confirming the change but not for initiating it. The authorization corresponds to the "ConfirmerAuthorization" attribute in the

properties for the block icon. The default setting is No. 6, Higher-level operator-process communications.

Note

The sections below describe the necessary transaction steps for the two operators. The figures illustrate the example of an F_REAL parameter with the login:

- level5 – Initiator
 - level6 – Confirmer
-

Note

When changing F_BOOL parameters using Safety Data Write, you must enter the value "true" or "false" and not "1" or "0". This entry is not case-sensitive.

Initiator: Initiating a change

1. Log on to the OS as a user with initiator authorization.
2. Click the desired block icon to open the faceplate.

The screenshot shows a dialog box titled "SDW_1/SDW_F_REAL". It contains several input fields and buttons. On the left, there are five rows of labels and input boxes: "Current Value" with "7", "Minimum" with "0", "Maximum" with "100", "Max Delta" with "10", and "Timeout" with "60" followed by the text "seconds". On the right, there is a label "New value" in red text next to an empty input box. At the bottom left is a button labeled "Test OS", and at the bottom right is a button labeled "Change".

The Safety Data Write dialog indicates the current value, the Timeout value in seconds, and, in the case of F_CHG_R, the values for the change limits (Minimum, Maximum, and MaxDelta) as well as the unit of measurement, where applicable.

3. Enter the new value in the "New value" field (using a maximum of 10 characters including decimal separators and plus or minus signs).

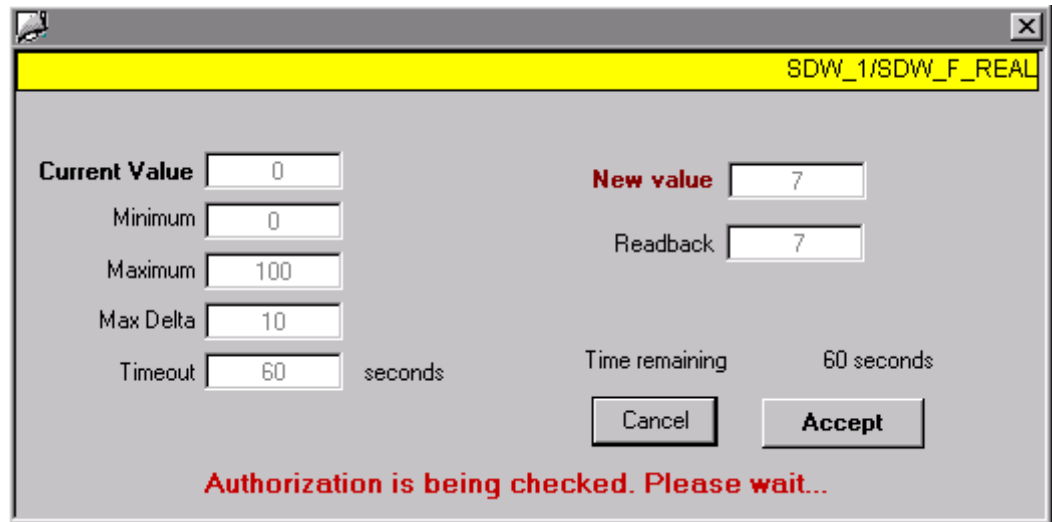
In the case of an F_REAL value, verify that the change limits (Minimum, Maximum, and MaxDelta) are not violated. If the new value violates one of the limit values, an error message is displayed and the "Change" button cannot be activated.

4. Click "Change". The modified value is also displayed in the "Readback" field.

11.3 Changing F-Parameters with Safety Data Write

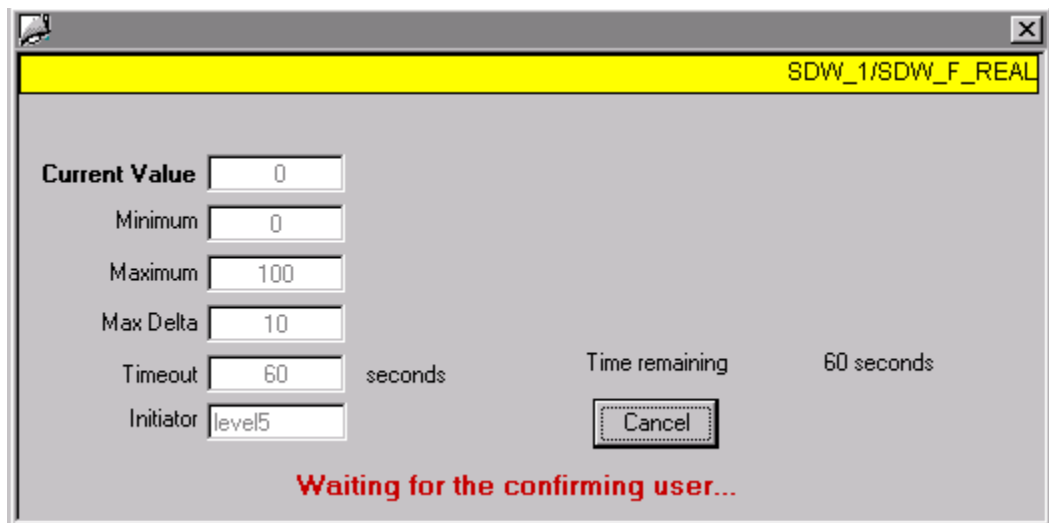
- 5. Compare the values in the "New value" and "Readback" fields. If they are identical, click the "Accept" button.

Note: If the block input EN_CHG changes to FALSE before you click the "Accept" button, this is indicated by a message, and the "Accept" button is disabled (see also the description of the F-blocks "F_CHG_R: Safety Data Write for F_REAL (Page 293)" and "F_CHG_BO: Safety Data Write for F_BOOL (Page 299)").



Result

The timeout counter is started and you are informed that the change must be confirmed by a second operator.



The confirmer must then continue the transaction.

If you cancel the transaction after clicking "Accept," check whether the previously valid value is displayed in the "Current value" field.

Confirmer: Confirming the change

Note

The confirmation must take place before the remaining time expires.

1. Log on to the OS as a user with "confirmer authorization".
You can log on to a second OS or on the same OS as the initiator.
2. Click the desired block icon to open the faceplate.

The screenshot shows a dialog box titled "SDW_1/SDW_F_REAL". It contains several input fields and buttons. On the left, there are fields for "Current Value" (0), "Minimum" (0), "Maximum" (100), "Max Delta" (10), "Timeout" (60) with "seconds" next to it, and "Initiator" (level5). On the right, there is a "Confirm value" field (7) and "Time remaining" (33) with "seconds" next to it. At the bottom right, there are "Cancel" and "Confirm" buttons.

3. Enter the new value in the "Confirm value" field. If the confirm value differs from the new value that was entered by the initiator, an error message is displayed and the "Confirm" button cannot be activated.

Note

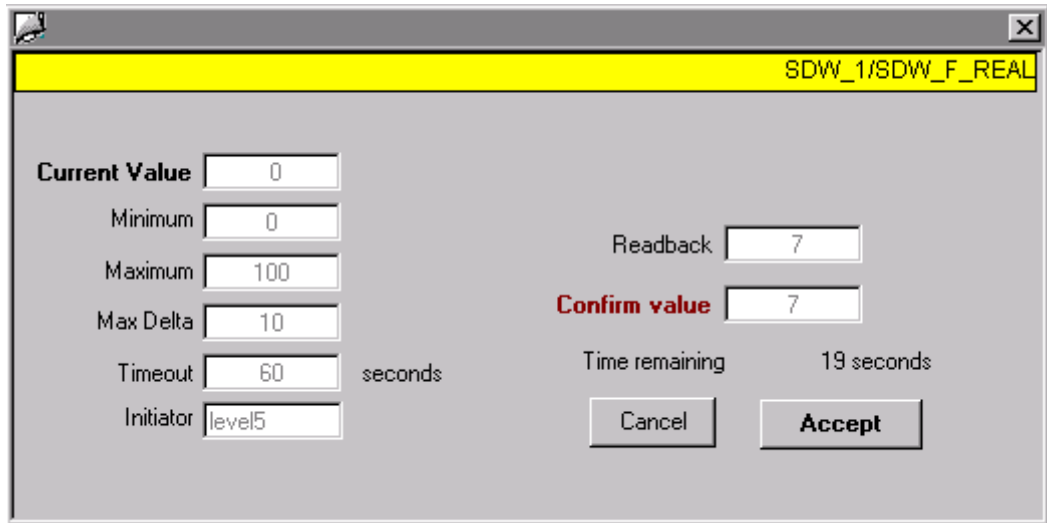
You must confirm the change by entering the new value separately. The value is deliberately not displayed since an unbiased confirmation by the second operator is required.

4. Click "Confirm".
The value entered by the initiator is displayed in the "Readback" field.
Note: If the block input EN_CHG is changed to FALSE, this is indicated by a message, and the input is canceled. Values can be re-entered once EN_CHG changes back to TRUE (see the description of the F-blocks "F_CHG_R: Safety Data Write for F_REAL (Page 293)" and "F_CHG_BO: Safety Data Write for F_BOOL (Page 299)").
5. Compare the values in the "Confirm value" and "Readback" fields. If they are identical, click the "Accept" button to permanently save the change. If the values do not match, you must click "Cancel".

Note: If the block input EN_CHG changes to FALSE before you click the "Accept" button, this is indicated by a message, and the "Accept" button is disabled (see also the

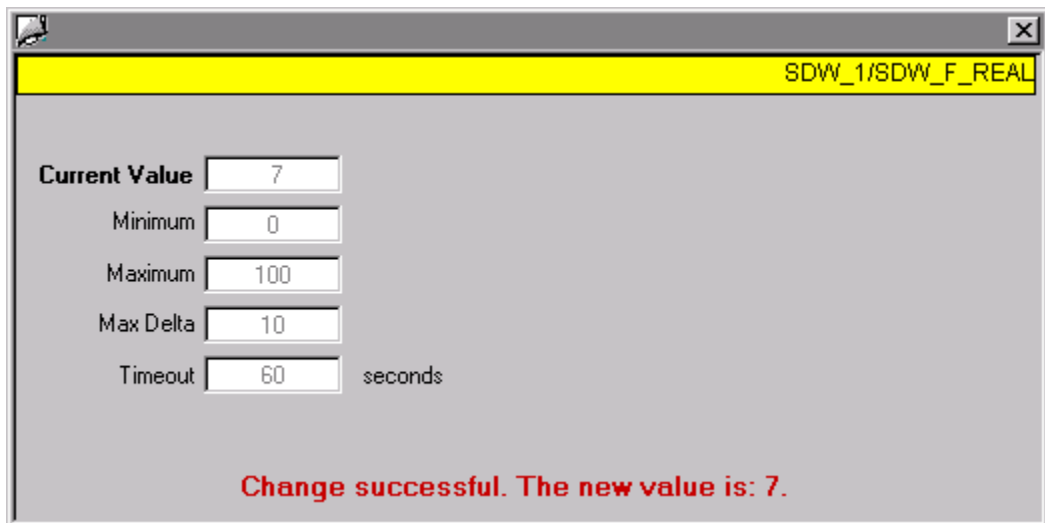
11.3 Changing F-Parameters with Safety Data Write

description of the F blocks "F_CHG_R: Safety Data Write for F_REAL (Page 293)" and "F_CHG_BO: Safety Data Write for F_BOOL (Page 299)".



Result

If the transaction is finished within the remaining time, a successful F-Parameter change is signaled.



11.3.3 Changing an F-Parameter with One Operator

Operator Authorization

If only one operator is to perform the transaction, this operator must be authorized to both initiate and confirm changes using Safety Data Write. The authorization must include the values of both the "InitiatorAuthorization" and "ConfirmerAuthorization" attributes. Default is No. 5, Operator-process communications and No. 6, Higher-level operator-process communications.

Transaction Sequence with Only One Operator

The procedure is the same as for operation with two operators, except that one operator is able to perform all of the steps (see also the section entitled "Changing an F-Parameter with Two Operators (Page 172)").

The difference is there is no waiting period for the confirmer. Rather, the operator is prompted immediately to enter the confirm value.

All other steps remain the same.

Compiling and commissioning an S7 program

12.1 Compiling an S7 program

Introduction

You compile a safety program by compiling the complete S7 program as usual in the *CFC Editor*.

Procedure

If an S7 program contains a safety program, this is automatically also compiled when the CFC charts are compiled. At the same time, fault-control measures are automatically added and additional safety-related checks are performed.

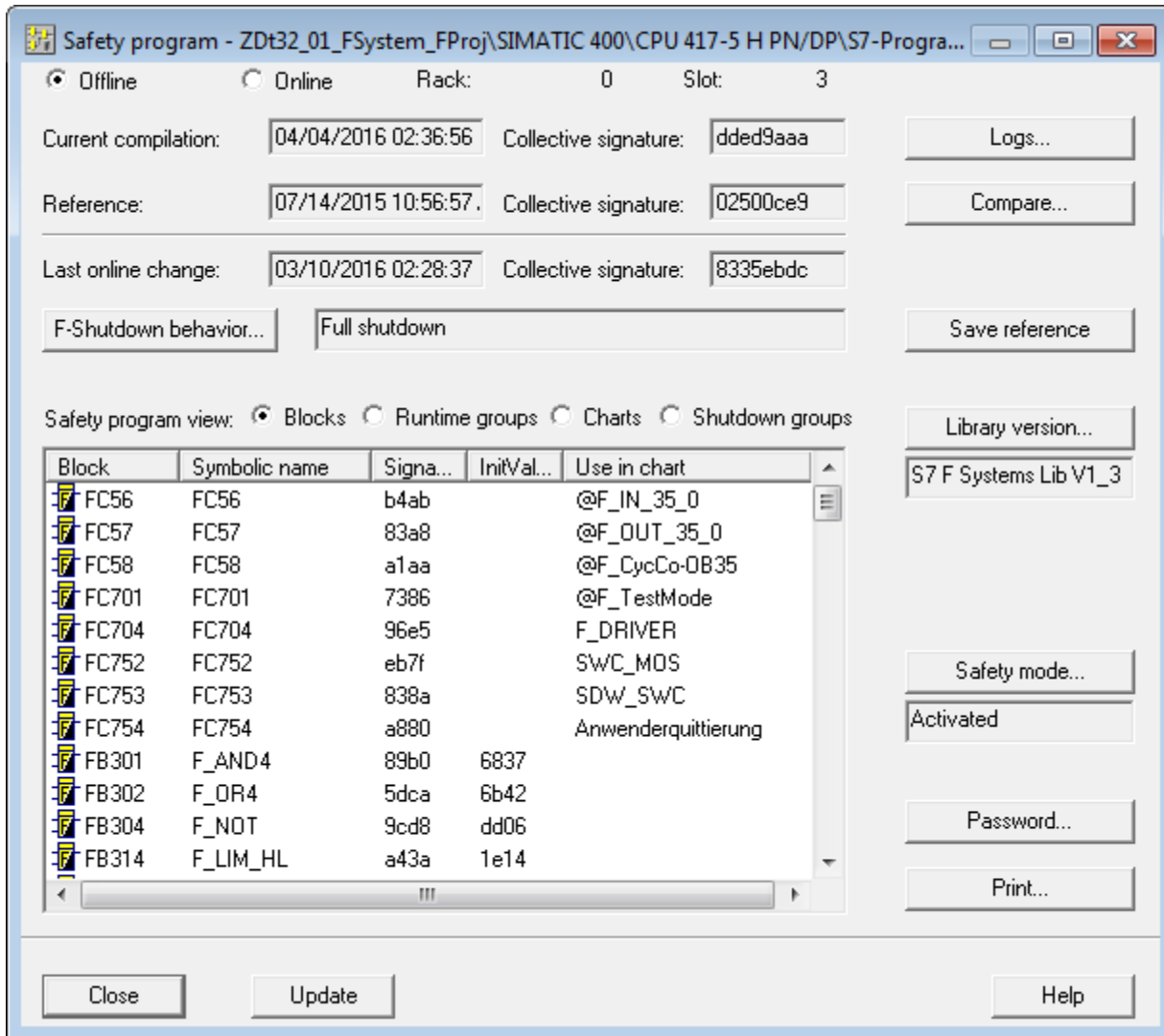
Observe the documentation for *CFC*: "SIMATIC Process Control System PCS 7 CFC for SIMATIC S7 (<https://support.industry.siemens.com/cs/ww/en/view/90683154>)".

If you have changed the safety program since it was last compiled, you will be prompted for the password of your safety program during the compilation operation. You must enter the password of your safety program to continue compiling.

12.2 "Safety Program" dialog

In *SIMATIC Manager*, open the "Safety Program" dialog by selecting the "Options > Edit safety program" menu command.

Starting from *S7 F Systems V6.2*, the size of the "Safety Program" dialog can be changed.



The following information about the safety program located online on the F-CPU or offline in the ES will be displayed in the "Safety Program" dialog box:

- A list of all included F blocks with signatures and initial value signatures
- Current compilation: Date and collective signature
- Reference: Date and collective signature
- Last online change: Date and collective signature

This data is provided for information purposes only and is not relevant for acceptance.

Buttons in the "Safety Program" dialog box

The dialogs you can access and the actions you can perform using the buttons in the "Safety Program" dialog box are described in the sections below.

| Button | Description |
|---------------------|--|
| F-Shutdown behavior | ""Shutdown Behavior" dialog box (Page 181)" |
| Logs | ""Logs..." button (Page 182)" |
| Compare | "Comparing safety programs (Page 185)" |
| Save Reference | ""Save Reference" button (Page 182)" |
| Library version | ""Library Version" button (Page 182)" |
| Safety mode | "Safety mode (Page 195)" |
| Password | ""Password for Safety Program Creation" dialog (Page 182)" |
| Print | "Printing project data of the safety program (Page 192)" |
| Refresh | ""Update" button (Page 184)" |

See also

Testing a safety program (Page 200)

12.2.1 "Shutdown Behavior" dialog box

Description

In the "Shutdown Behavior" dialog box, you can choose how the safety program should behave when an error is detected, i.e., during an F-STOP:

- "Full shutdown": All F-Shutdown groups of a safety program are shut down the first time an error is detected in an F-Shutdown group.
- "According to the configuration of F_SHUTDN":
 - The faulty F-Shutdown group or groups of a safety program are shut down the first time an error is detected in an F-Shutdown group (partial shutdown).

or

 - All F-Shutdown groups of a safety program are shut down the first time an error is detected in an F-Shutdown group.

You must recompile the S7 program after changing the shutdown behavior.

You must also enter the password for the safety program when you change the shutdown behavior.

See also

F-STOP (Page 93)

12.2.2 "Logs..." button

Click the "Logs..." button to open the "Logs" dialog of the *CFC Editor*. The "Compile" and "Download" logs are relevant for the safety program acceptance test. For information about the acceptance test, refer to the section entitled " System Acceptance Test (Page 213) ".

12.2.3 "Save Reference" button

You can save all data of a safety program (charts, parameters, etc.) as a reference to be used as necessary for comparisons.

12.2.4 "Library Version" button

Description

The "Library Version..." button enables you to upgrade the F-Library version used in the project to the current version of the F-Library.

The window below the button displays the F-Library version *currently used in the project*.

12.2.5 "Password for Safety Program Creation" dialog

Description

In the "Create password for safety program" dialog, you can create a new password or change an existing password for the safety program.

Target system and program name:

The upper part of the dialog shows the target system and program name for the safety program for which the password is being created or changed.

Overview

You must create a password for each safety program. You must enter this password using the "Password..." button in the "Safety program" dialog before you can perform the actions described in section "Overview of access protection (Page 69)".

When the password for the safety program is entered for one of these actions, the user obtains access authorization. This access authorization is valid for one hour. After this time elapses, the user is prompted to enter the password again when he wants to perform one of the above-named actions.

The access authorization is reset to 1 hour following each safety-related action.

The access authorization can also be canceled in this dialog.

"Increased password security" check box

With this option you can activate the "Increased password security" in order to use a more secure password.

- If you activate the option, the rules corresponding to the description "Criteria for a secure password" below apply to the password.
- If you clear the option, the previous rules apply to the assignment of the password.

Criteria for a secure password

To ensure a secure password, it must meet the following criteria when created for the first time or changed:

- Password length: at least 8, maximum of 32 characters
- At least one upper case letter of the Latin alphabet (A - Z); also diacritical marks (umlauts and letters with accents)
- At least one lower case letter of the Latin alphabet (a - z); also ß and diacritical marks (umlauts and letters with accents)
- At least one number (0-9)
- At least one of the following special characters:

~ ! @ # \$ % ^ & * _ - + = ` | \ () { } [] : ; ' " < > , . ? /

These criteria apply when the "Increased password security" option is activated in the "Create password for safety program" dialog.

Creating a new password

During the initial setup of a new password, select the password in conformance with the criteria described below and enter it in the "New password" and "Reenter password" fields. In this case, the "Old password" field is deactivated.

By selecting the "Increased password security" check box, you can use a more secure password that conforms to the description "Criteria for a secure password" above.

If a password has not yet been created for the safety program, you will be prompted to do so if a password is required for the desired configuring task, e.g. when inserting an F-block in a CFC chart or when inserting fail-safe modules in HW Config.

You can find additional information on the password prompt in section "Overview of access protection (Page 69)" in the "Password for safety program" table.

Changing a password

To change a password, you must enter the old password in the "Old password" field.

Then, choose the new password based on whether or not the "Increased password security" check box is selected and enter it in the "New password" and "Reenter password" fields.

Revoking access permission

You can use the "Logout" button in the "Access permission" area to revoke the 1-hour access permission period since the last time the password was entered.

Any user who then wants to perform an action that requires entry of a password must now enter the password for the safety program again.

12.2.6 "Update" button

Description

You can use this button to update all displayed information. This can be necessary, for example, when changes were made in other applications, *such as the CFC Editor* since the dialog was opened.

12.3 Comparing safety programs

Introduction

The "Compare Programs" dialog box enables you to compare safety programs and display and print out differences.

You can compare the following safety programs:

- Online safety program in the F-CPU
- Current offline safety program
- Last compilation of the current S7 program
- Saved reference program
- Other project

The result of the comparison shows you whether the following are the same or different:

- Collective signature
- Individual signatures
- Parameter values
- Differences in the safety program and control structures
- Modified or deleted F-Blocks and interconnections, etc.

With the "Compare Programs" dialog, you can also tell if a safety program was *not* modified. For this purpose, compare the safety program with the reference program.

Starting from *S7 F Systems V6.2*, the dialog can be resized to make the table easier to read.

In *S7 F/FH Systems V6.1* and later, system-related changes are shown in a combined display, making it easy for you to identify changes that are relevant for checks. This facilitates the acceptance test for changes.

System-related changes are primarily found:

- In system charts beginning with @F_x
- In runtime groups beginning with @F_x
- On driver blocks

Program/reference

Select one of these option boxes to specify whether you want to compare the current program or the reference program.

Compare with:

Use this drop-down list box to specify the second safety program to which you want to compare the safety program you just selected.

| Program | Compare with ... | |
|------------------|-------------------------|--|
| | Reference | Last saved reference for this safety program |
| | Last compilation | The last compilation of this S7 program during which safety-related changes were detected. |
| | Online | Currently downloaded safety program in the F-CPU |
| | Other project | Any offline program. Use the "Browse" button to select the offline program. |
| Reference | Compare with ... | |
| | Current safety program | Current offline program |
| | Last compilation | The last compilation of this S7 program during which safety-related changes were detected. |
| | Online | Currently downloaded safety program in the F-CPU |
| | Other project | Any offline program. Use the "Browse" button to select the offline program. |

"Browse" button

Use this button and the "Open" dialog to select the offline program of any project to be compared.

"Start" button

Click this button to start the comparison.

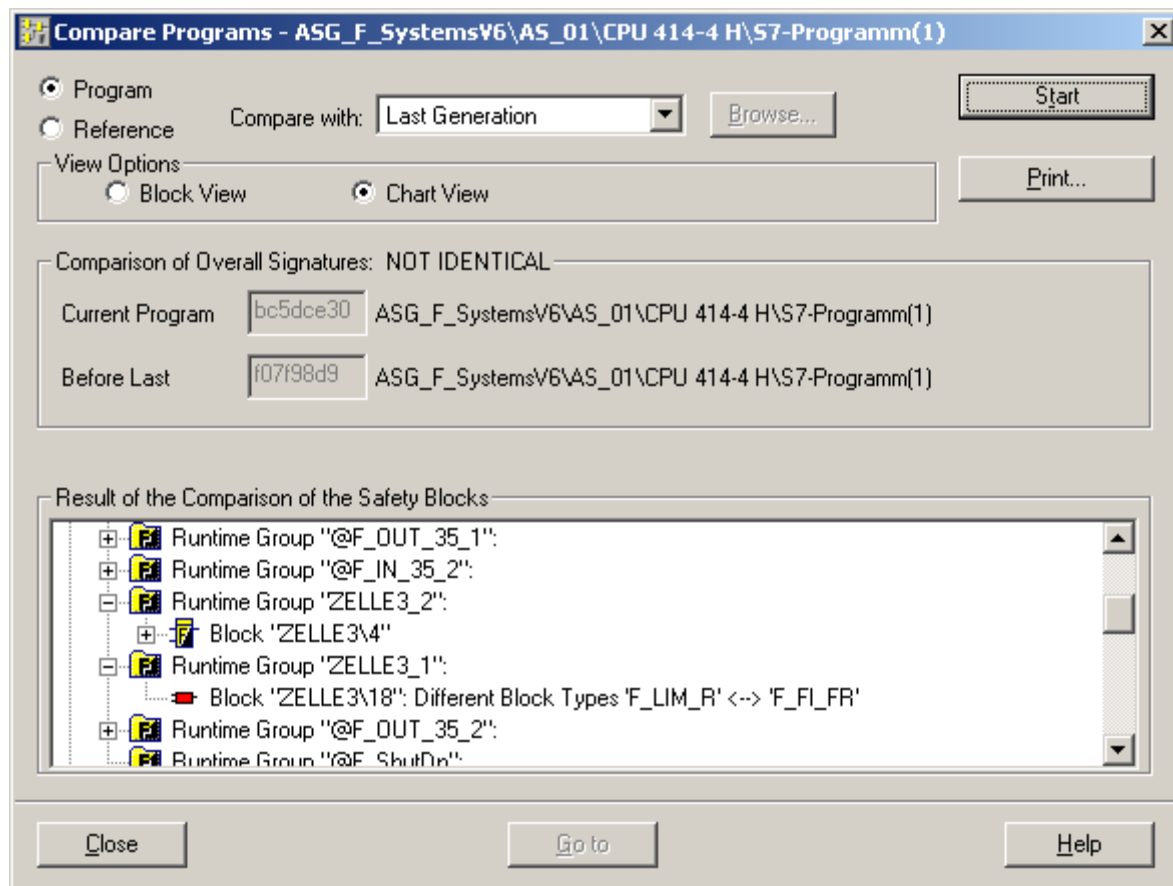
View options

If you want to compare two offline programs, you can switch back and forth between the following options by clicking the relevant option buttons:

- **Block view:**
Shows you a list with the differing blocks (different block signatures).
- **Chart view:**
Shows you a hierarchy of all differences in the:
 - Task
 - F-Runtime group
 - F-Block
 - Parameters

In this view, the "Go to" button is available.

Result of the comparison (both safety programs offline)



A note is displayed indicating whether or not the collective signatures of all F-Blocks are identical.

Display of differences in the block view

In the block view, all F-blocks whose signatures have changed are displayed with the relevant signature, but the F-Runtime group and task are not displayed.

Display of differences in the chart view

The differences between charts are displayed in a hierarchical format similar to Explorer. In this view, all F-Blocks are shown under the relevant task and F-Runtime group. Information about the possible changes are shown individually for each F-Block. This information relates to the task, the F-Runtime group, and the sequence within the F-Runtime group, as well as the parameter assignment and interconnections of the F-Blocks.

Only tasks, F-Runtime groups, F-Blocks, and parameters in which changes were found are displayed.

Changes are described as follows:

| Text | Meaning |
|--------------------------|---|
| Deleted | F-Block only present in source |
| Added | F-Block only present in comparison program |
| Runtime position changed | F-Block is located in a different runtime position in the F-Runtime group |
| Interface changed | <ul style="list-style-type: none"> • Additional parameters • Removed parameters • Modified data type (e.g. F-Bool <- Bool) |
| Signature changed | Signature of F-Block type (FB) changed |
| Value: "new" <- "old" | The parameter assignment of an input or output or the interconnection source of an input has been changed from "old" to "new". "Not-interconnected" can also be specified as the interconnection source if an interconnection has been deleted or newly created. |

Note

If "Different versions of the F-reference data" appears in the chart view when comparing the safety program to a reference, this means that you created the reference with an older version of *S7 F Systems* and did not overwrite it with the current version during migration.

Instead, use the old project version that you archived prior to migration.

Displayed changes

Note the following when changing names:

The *F Systems* comparator references the elements according to name. If an element name is changed, the element can no longer be assigned.

- Chart names
- Name of a runtime group
- Block name (instance in a chart)
- Parameter name (for F-Block types)

Although chart names are not relevant for runtime, changes still affect the "Chart view":

- Each time a chart name is changed, the chart is displayed with the old name as "Deleted" and with the new name as "Added".
- In *CFC*, an F-Runtime group with the same name is renamed at the same time. Therefore, this F-Runtime group is also displayed with the old name as "Deleted" and with the new name as "Added".
- All interconnects of F-Blocks outside of this chart to F-Blocks within this chart as displayed as changed. The reason for this is that the chart name is also used as the name component of an interconnection peer to identify the interconnection.
- The block view correctly returns no difference in this case. Likewise, the collective signature of the safety program does not change. In order to prevent such unnecessary

entries in the chart view, we recommend that you do not rename any F-Charts or shift between F-Charts after performing the acceptance test.

Note the following:

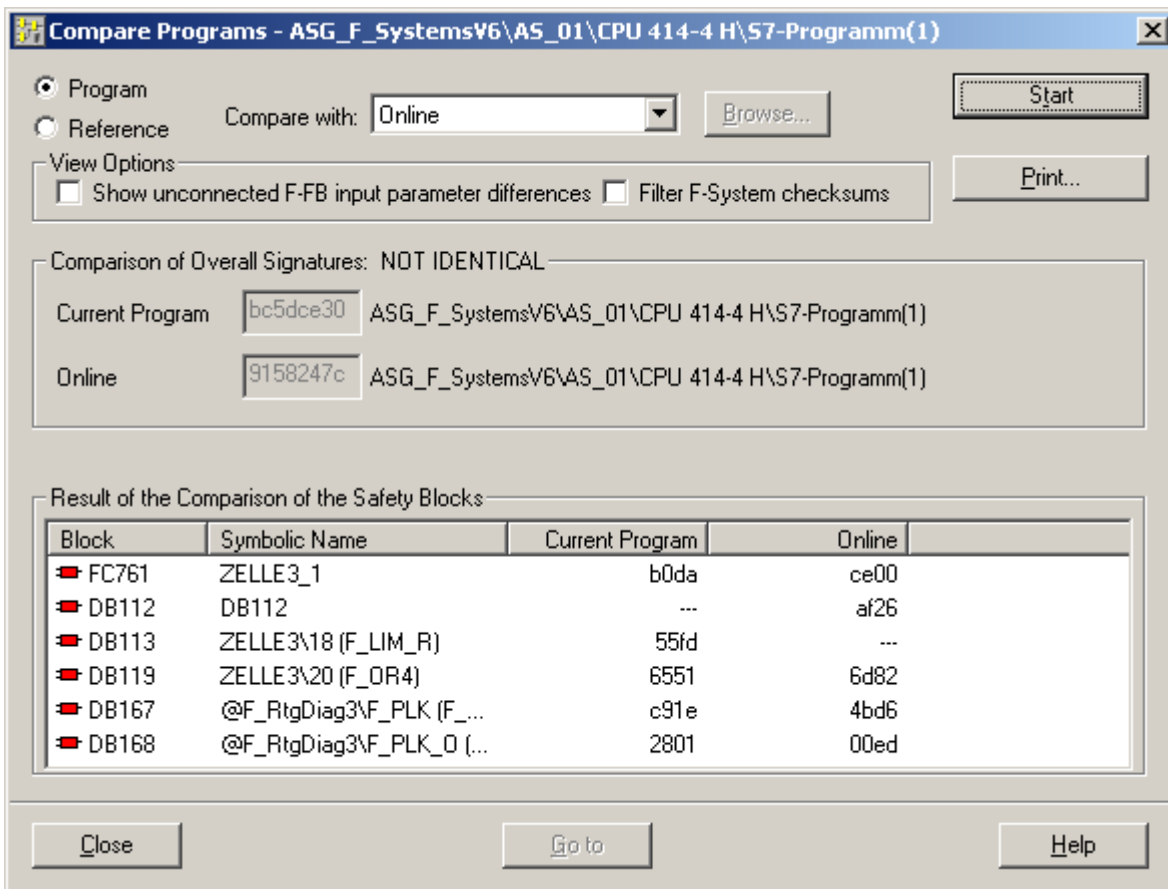
- In the chart view of the comparison, only differences pertaining to the safety program are generally displayed. In particular, changes in interconnections between the safety program and the standard program or global addresses are not displayed.
- If an interconnection of an output is changed at the same time as the initial value of this output, the modified interconnection will be displayed, but not the modified initial value.

Result of the comparison (online safety programs with offline)

When a comparison to the online program is made, an indication is given as to whether the source, load memory, and work memory match (this allows you to detect non-permissible data manipulations on non-interconnected, fail-safe input parameters in the work memory).

If you have selected the online program in the "Compare with" drop-down list box, only the block view is available. In this case, the following two view options are available:

- Show unconnected F-FB input parameter differences
- Filter F-System checksums



Just as in the offline block view, the window shows you all F-blocks whose signatures differ.

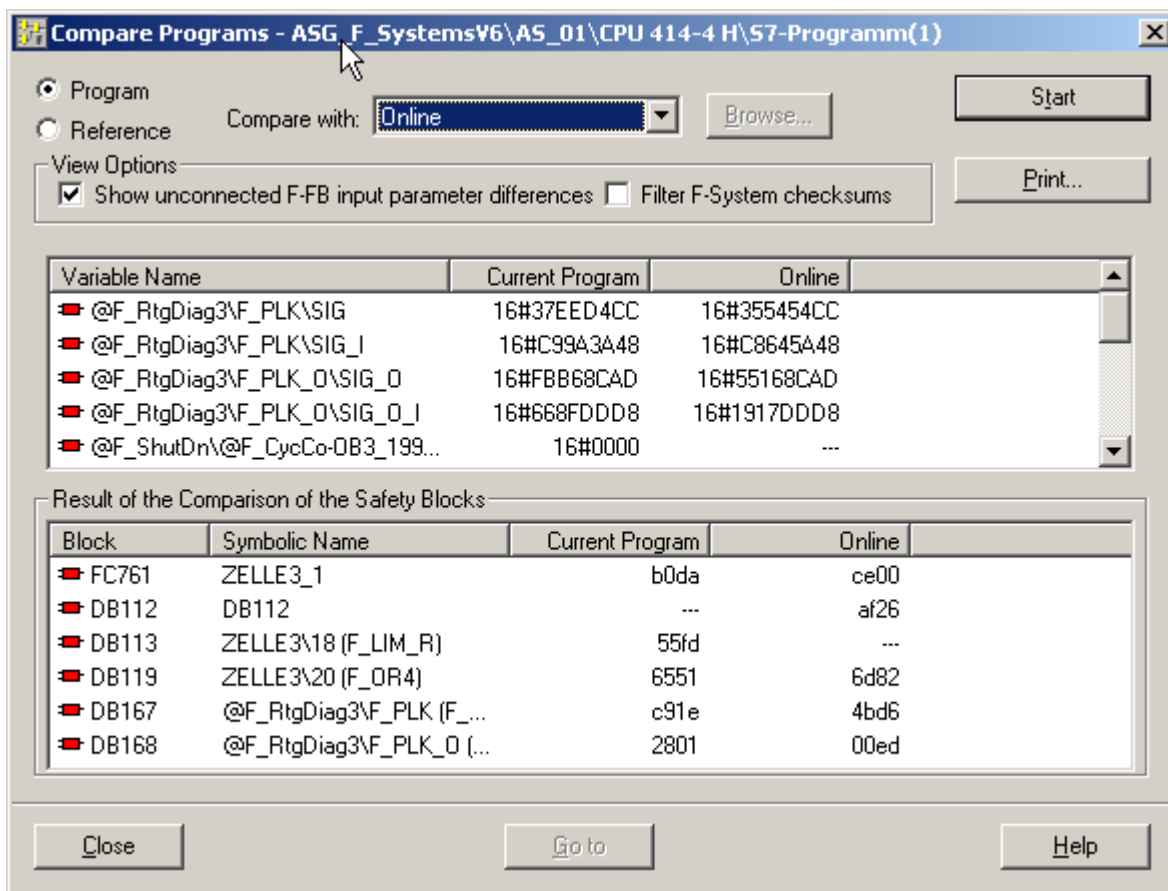
"Show unconnected F-FB input parameter differences" view option

This option compares the assigned parameter values of all non-interconnected inputs. It compares the online program to the offline program.

The differences are displayed in the list at the top of the dialog box.

This view option is normally selected only if the collective signatures already match. This indicates that the offline program has not been changed since the last time it was downloaded to the F-CPU.

This option enables you to perform a thorough search for parameters that have been changed online, but not through compilation or download.



"Filter F-System checksums" view option:

This option suppresses expected differences that can occur when the F-CPU writes to specific F-Blocks (for example, input signature values of F_PLK and F_PLK_O). You can only use this view option in connection with the "Show unconnected F-FB input parameter differences..." option.

"Print" button

Click this button to print out the result of the comparison.

"Go to" button

In the chart view, you can select any F-Block or parameter in the differences display and then click this button to access the relevant block in the *CFC Editor*.

See also

Migration to S7 F Systems V6.2 (Page 32)

12.4 Printing project data of the safety program

Requirement

The safety protocol can be printed in landscape format.

To ensure that all columns are printed, make the following settings:

1. In SIMATIC Manager, select the menu command **File > Page Setup....** In the following dialog, select landscape format in the "Paper Size" tab.
2. Also select landscape in the format settings of the printer or the PDF generator.

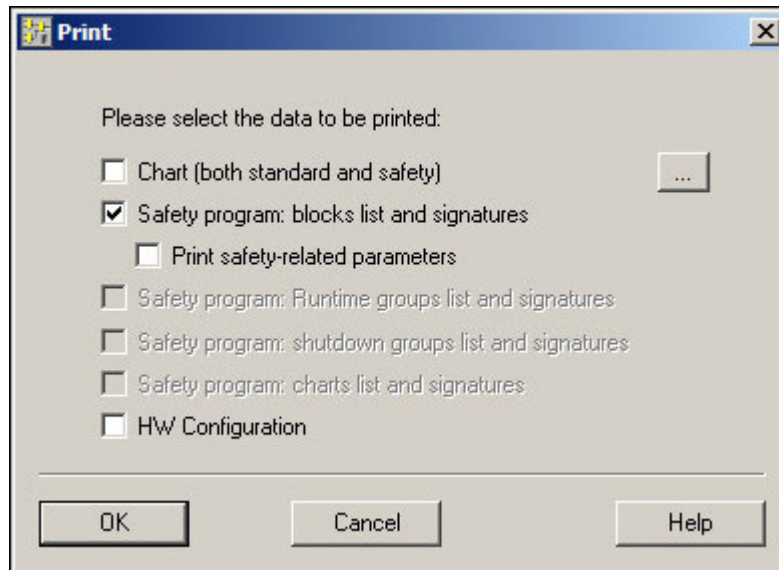
Procedure:

You receive a printout of all important project data as follows.

1. Select the program folder (e.g., "S7 Program").
2. Select the menu command **Options > Edit safety program.**

The "Safety Program" dialog will appear.

3. Click "Print". In the "Print" dialog, you can select the parts of the project you want to print:



– **Chart (both standard and safety):**

Prints all or selected charts of the standard program and safety program in a graphical representation.

A selection dialog for selecting the charts to be printed can be opened using the "..." button. Additional information can be found in the next section "Selecting the charts for printing".

If not all charts were selected in the selection dialog, the "Chart (both standard and safety)" check box is shown partially activated.

– **Safety program: Block list and signatures**

Offline/online status log

Name of the safety program

Date of the last compile operation and the collective signature of the safety program

Date of the last compile operation and collective signature of the reference program

F-blocks in the safety program

Print safety-related parameters

The footer on each page of the printout shows you the version of *S7 F Systems* used to generate the printout along with the collective signature.

– **HW configuration:**

Printout of the complete hardware configuration or portions thereof. The "Print" dialog will appear so that you can specify what information is to be printed for the F-I/O.

The printout of the safety program also contains the collective signature and the date of the last compilation, which are relevant to the onsite acceptance test of the safety program (e.g., by experts). The collective signature of the compiled S7 program appears twice in the printout:

1. In the program information section as a value of the block container
2. In the footer as a value from the chart container

(See also section "Checking the signatures").

Selecting the charts for printing

In the "Select charts" dialog, you can select charts for printing.

To open this dialog, open the "Print" dialog using the "Print" button in the "Safety program" dialog and then click the "..." button of the "Chart (both standard and safety)" option.

Structure of the dialog:

- "Target system"

The output field shows the target system from which the charts are selected.

- "Program name"

The output field shows the path and the name of the S7 program from which the charts are selected.

- "Filter" drop-down list
A variety of options are available in the drop-down list for filtering the charts.
- "Display" entry field
In this field, you can enter one or more characters to be searched for in the currently displayed chart names in the table. The filter result is immediately displayed in the table.
- Table:
The table shows the charts available for selection. The content of the table is influenced by the "Filter" drop-down list and the "Display" entry field.
 - Only the selected charts are printed, i.e. the charts for which the check box in front of the chart name is selected.
 - Functions in the shortcut menu of a table row, e.g. "Invert selection", and the shortcut "<Ctrl>+<A>" are available for selection.
 - The width of the table columns can be changed in the table header.

12.5 Safety mode

Introduction

Safety mode of the safety program in the F-CPU can be deactivated and reactivated at times. This allows you to make changes in the safety program in RUN mode.

Description

All the safety mechanisms for fault detection and fault reaction are activated in safety mode. The safety program cannot be modified during operation (in RUN mode) in safety mode.

You can activate or deactivate safety mode in the F-CPU in RUN mode using the "Safety Mode..." button in the "Safety Program" dialog. Downloading safety program changes in RUN mode is only made possible by temporarily switching the safety mode to "deactivated" using this button.

The window below this button indicates whether safety mode is "activated" or "deactivated". It will indicate "Unknown" if the safety program does not correspond to the safety program in the F-CPU or if no communication is taking place with the F-CPU.

You can also determine whether or not safety mode is enabled from the SAFE_M output of the F_SHUTDOWN block (located in the @F_ShutDn chart).


See also

Downloading the safety program (Page 198)

12.5.1 Deactivating safety mode

Introduction

Deactivation of safety mode enables changes to be made to the safety program during operation (RUN). For this purpose, mechanisms for detecting changes to the safety program that would trigger shutdown of the safety program and its outputs in activated safety mode are deactivated. The safety program and thus the programmed safety functions continue to be executed. "Incidental hardware faults" will continue to be detected and the diagnostics of the modules remain active.

| |
|--|
|  WARNING |
| Deactivating safety mode |
| <p>Because changes can be made to the safety program in RUN mode when safety mode is deactivated, you must observe the following:</p> |
| <ul style="list-style-type: none">• Deactivation of safety mode is intended for test purposes, commissioning, etc. Whenever safety mode is deactivated, the safety of the plant must be ensured by other organizational measures, such as operation monitoring and manual safety shutdown.• Deactivation of safety mode must be verifiable. Logging is required and can, for example, be guaranteed by using an OS. The automatically placed F_SHUTDOWN block generates corresponding messages for this. Otherwise, you must log the deactivation of safety mode through organizational measures.• Furthermore, we recommend that deactivation of safety mode be displayed, e.g. on the OS. For this purpose, the automatically placed F-block F_SHUTDOWN sets the SAFE_M output to "0" when safety mode is deactivated (or F-block F_TESTM sets the TEST output to 1).• Safety mode is deactivated only F-CPU-wide. For this reason, you must observe the following for safety-related CPU-CPU communication: If the F-CPU with the F_SENDBO, F_SENDR or F_SDS_BO is in deactivated safety mode, you can no longer assume that the data sent by this F-CPU were generated safely. To ensure the safety of the parts of the plant influenced by the sent data, you must then also take organizational measures, e.g. monitored operation and manual safety shutdown, or output safe fail-safe values instead of the received data in the F-CPU with the F_RCVBO, F_RCVR or F_RDS_BO through evaluation of SENDMODE. |

Requirements

The F-CPU is in RUN mode and safety mode is activated.

Procedure

1. Select the F-CPU or its S7 program in *SIMATIC Manager*.
2. Select the menu command **Options > Edit safety program**.
3. Select the "Safety mode" button.

You can then download changes in the safety program to the F-CPU during operation (in RUN mode).

12.5.2 Activating safety mode

Introduction

After changes in the safety program are downloaded, you must reactivate safety mode in order to guarantee safe execution of the safety program.

Requirements

The F-CPU is in RUN mode and safety mode is deactivated.

Procedure

1. Select the F-CPU or its S7 program in *SIMATIC Manager*.
2. Select the menu command **Options > Edit safety program** .
3. Select the "Safety mode" button.

Note

If the safety program detects a safety-related error during deactivated safety mode, it is no longer possible to activate safety mode. You then receive a corresponding message with corrective actions.

See also

Downloading changes (Page 203)

12.6 Downloading the safety program

Introduction

Following compilation, you can download the CFC program to the target system. Depending on whether safety mode is activated or deactivated, you can download the entire safety program or changes in the safety program as follows:

| Downloading | F-CPU in STOP | F-CPU in RUN, safety mode activated | F-CPU in RUN, safety mode deactivated |
|--------------------------------------|---------------|--|--|
| Entire S7 program | Possible | F-CPU is automatically put in STOP mode by the <i>CFC Editor</i> | F-CPU is automatically put in STOP mode by the <i>CFC Editor</i> |
| Changes in the standard user program | Possible | Possible | Possible |
| Changes in the entire S7 program | Possible | Not possible | Possible |


Requirements

- The hardware configuration data of the station is downloaded to the F-CPU
- The S7 program was compiled without error.
- You have access rights to the target system.
- An online connection exists between the F-CPU and your ES.

Rules for downloading

- You can only download the safety program from the *CFC Editor* or from the *SIMATIC Manager* via the chart folder.
- When an accepted safety program is downloaded, you must check the collective signature after downloading the same as for acceptance.

See also "Collective signature" in section " Downloading the S7 program to the F-CPU (Page 219) ".

| |
|--|
| <p> WARNING</p> <p>Do not copy F-blocks with <i>SIMATIC Manager</i></p> <p>As is usual in <i>PCS 7</i>, you must not copy individual blocks between block containers online and offline. Use the download function in the <i>CFC Editor</i> for this or download the chart folder.</p> <p>You can find detailed information in the "SIMATIC Process Control System PCS 7 CFC for SIMATIC S7 (https://support.industry.siemens.com/cs/ww/en/view/90683154)" manual, sections "Downloading" and "Reading back charts"."</p> |
|--|

12.6.1 Downloading the S7 program

Procedure

To download the safety program to the target system, select the menu command **CPU > Download > Entire program** in the *CFC Editor*. The F-CPU is thereby set to STOP.

Note

Before the safety program is downloaded, a prompt for the password of the F-CPU is displayed when changes in the safety program are detected.

Working with safety programs on memory card

WARNING

Safety program on a memory card

If you are using the safety program on a memory card, you must observe the following:

- Before you switch the S7 F System to RUN mode, compare the collective signature of the safety program on the Flash EPROM memory card with the collective signature of the reference data. If necessary, mark the memory card with the collective signature.
- For a fault-tolerant S7 FH System, you ensure that the memory cards of the redundant F-CPU's are of the same type (RAM or Flash EPROM) and redundant Flash EPROM memory cards contain the same safety program.
- You ensure access protection with regard to removal and insertion of memory cards.

WARNING

If multiple F-CPU's can be reached from an ES via a network (e.g. MPI), you must take the following additional measures to ensure that the safety program is downloaded to the correct F-CPU.

Use F-CPU-specific passwords, e.g. a password for the F-CPU's with appended MPI address "FCPU PW_8". The password has a maximum of 8 characters, including at least one special character. In STEP 7 V5.5.4 HF9 and higher, the password must contain 8 characters for new projects.

Note the following:

- Before a safety program for which access permission by means of an F-CPU password does not yet exist is downloaded to an F-CPU, any existing access permission for another F-CPU must first be canceled.

12.7 Testing a safety program

Introduction

Testing occurs as usual in *CFC* by switching to test mode.

Switching to test mode

After compiling and downloading, you have the option of testing the safety program. You test safety programs by switching to test mode using the **Debug > Test Mode** menu command in the *CFC Editor*. In test mode, you have an online connection to the automation system (F-CPU).

Rules for testing



WARNING

Shutdown of the safety program following changes to fail-safe outputs

In test mode of the *CFC Editor*, you can monitor safety programs and modify inputs of F-Blocks that are not interconnected. It is not permitted to change fail-safe outputs and automatically initialized inputs/outputs online; this causes the safety program to shut down.

12.7.1 Testing with S7-PLCSIM

Procedure

The *S7-PLCSIM* software package allows you to simulate a safety program on your ES.

For simulation of your safety program with *S7-PLCSIM*, use the same procedure as in the standard case.

If you download the safety program in *S7-PLCSIM*, the "Set up access authorization" dialog appears. You will be prompted for the password for the F-CPU.

You can only download changes in the safety program with the complete safety program.

Note

If an F-STOP is triggered for the safety program, you must then follow the procedure below:

- Perform a memory reset for the virtual F-CPU (*S7-PLCSIM*).
 - Download the configuration data and the S7 program again.
-

 **WARNING**

A simulation is no substitute for a function test!

If the simulation takes place on an ES with an online connection to the F-CPU, you must not deactivate safety mode. In addition, you are not permitted to have access authorization through the password for the F-CPU.

12.8 Modifying a safety program

Introduction

Changes in the safety program can be made offline as well as online. Online changes are made by means of the CFC test mode and take effect immediately. You must then download offline changes to the F-CPU.

Note

Safety program changes made otherwise, for example, by means of the "Monitor/Modify Variables" function, can lead to an F-STOP.

12.8.1 Online changes in CFC test mode

Introduction

In test mode of the *CFC-Editors*, you have the option of changing the values of non-interconnected inputs of F-blocks during operation.

Rules

- For inputs in safety data format, you may only change the DATA component and not the COMPLEM or PARID component.
- You are not permitted to change outputs or any inputs not documented in the block description.

Requirements

Ensure that the following requirements are met before you switch on test mode of the *CFC Editor*.


- The F-CPU must be in RUN mode.
- Safety mode of the safety program must be deactivated. Otherwise, you will be prompted to deactivate safety mode when you attempt to change the first parameter.

| |
|---|
|  WARNING |
| Change of the collective signature following changes in CFC test mode |
| Changing the safety program in CFC test mode causes the collective signature to change. This means that the safety program must undergo acceptance again, if necessary. |

Procedure

For changing the fail-safe block I/O, follow the usual procedure in the *CFC Editor*.

The collective signature at the F_SIG_OUT output of the F_SHUTDN F-block is set to 0 at the first change in CFC test mode and updated after CFC test mode is ended.

| |
|--|
|  WARNING |
| Do not change values created during compilation |
| When safety mode is activated, direct operator control of safety programs is not permitted! You may input safety parameters for non-interconnected inputs: |
| <ul style="list-style-type: none">• from the standard user program via F-conversion blocks with additional validity check <i>or</i>• in test mode of the <i>CFC Editor</i> and with deactivated safety mode <i>or</i>• with the "Safety Data Write" or "Secure Write Command++" function |
| Failure to observe this warning will trigger an F-STOP. The following diagnostics event is then entered in the diagnostics buffer of the F-CPU: |
| <ul style="list-style-type: none">• "Safety program: Error detected" (event ID 16#75E1) |

12.8.2 Downloading changes

Requirements

- Safety mode must be deactivated.
- S7 FH Systems must be in redundant system state.

Procedure

1. To download changes in the safety program, follow the usual procedure for downloading changes in *CFC*. For more information, refer to the "SIMATIC Process Control System PCS 7 CFC for SIMATIC S7 (<https://support.industry.siemens.com/cs/ww/en/view/90683154>)" manual.
2. Activate safety mode again by responding to the prompt that appears.
3. If necessary, repeat Steps 1 and 2 to download incremental changes, for example.
4. In the *SIMATIC Manager* select the menu command **Options > Edit safety program**.
5. Follow the procedure described in Section "Acceptance test of safety program changes (Page 221)".

 **WARNING**

Abort of download operation

If the download operation is aborted, you must repeat the download and the check of the collective signature online and offline. In this way, you ensure the consistency of data in the load memory and work memory.

Note

Undoing changes

If you undo a change and download it nevertheless, it is possible that a different collective signature will be generated than before the change.

 **WARNING**

Moving F-blocks or F-runtime groups

Note that

- F-blocks that were moved to another F-runtime group
- or*
- F-runtime groups that were moved to another task

while downloading changes over multiple processing cycles may be processed multiple times or not at all.

 **WARNING**

Modifying the safety program in RUN mode

- When changes are made to the safety program in RUN mode when safety mode is deactivated, switchover effects may occur. Take additional organization measures to ensure that this does not impair the safety of the plant.
- Whenever possible, the standard user program and the safety program should be changed separately and the changes downloaded. Otherwise, an error may be downloaded to the standard user program while the required protection function in the safety program is not yet effective or switchover effects may occur in both programs.

Note

- Note also the corresponding FAQs (<http://support.automation.siemens.com/WW/view/en/13711209/133000>) on the Internet regarding downloading changes.
- Changes to the automatically generated charts and F-runtime groups are generally forbidden and may trigger an F-STOP. Exceptions:
 - The MAX_CYC parameter of the F_CYC_CO blocks for which you assign the F-monitoring time for a cyclic interrupt OB
 - Parameter assignments for the F_SHUTDN block for the F-shutdown behavior

Note

Splitting or combining F-runtime groups when safety programs are running represents an essential change in the run sequence. Before downloading changes with the "Compare safety programs" dialog, check for moved F-module drivers.

This can lead to the following unintended behavior when changes are downloaded in RUN mode:

- Passivation of output channels
- Processing of outdated input data at the input channels

The change in the run sequence causes the associated F-module drivers to be moved to other F-runtime groups.

12.8.2.1 Changes that can be transferred by downloading changes

You can transfer the following changes to the F-CPU by downloading changes.

If you do not observe the information in Chapter "Downloading changes (Page 203)" and the boundary conditions listed below, an F-STOP can be triggered for the safety program.

- Inserting new F-Runtime groups with new instances of F-Blocks/F-Block types.
- Inserting, modifying, and deleting interconnections of F-Blocks.
- Deleting and reinserting F-Blocks or moving F-Blocks in the runtime sequence within the F-Runtime group.
- Changing values of inputs and outputs of F-Blocks.

Exception: Changes in safety-related communication between F-CPU's (see " Change in the safety-related communication between F-CPU's (Page 208) ")

- Moving of instances of F-Blocks/F-Block types between F-Runtime groups within an F-Shutdown group.
- Moving of instances of F-Blocks between F-Runtime groups of different F-Shutdown groups.

Boundary condition: Note that all fail-safe channel drivers of an F-I/O must be contained in a common F-Shutdown group.

- Inserting/deleting F-Shutdown groups by means of F_PSG_M

Boundary condition:

- There must be no instances of F-Block types prior to the position in the F-Shutdown group where you insert or delete the F_PSG_M.
- Note that all fail-safe channel drivers of an F-I/O must be contained in a common F-Shutdown group.

- Moving the F-Runtime groups that do not contain instances of F-Block types to another task.

Boundary conditions:

- Note that all fail-safe channel drivers of an F-I/O must be contained in a common F-Shutdown group.

- Adding F-I/O by means of CiR

Boundary condition: Note the information about CIR in Chapter " System modifications during operation (Page 64) ".

12.8.2.2 Changes requiring an F-Startup

The following changes require an F-Startup of the safety program. You cannot download these changes to the F-CPU without triggering an F-STOP; see the section entitled " F-STOP (Page 93) ". These changes may only be downloaded by means of a complete download.

- Dividing/combining F-Shutdown groups by means of F_PSG_M
 - There are instances of F-Block types prior to the position in the F-Shutdown group where you insert or delete the F_PSG_M.
- Moving of instances of F-Block types between different F-Shutdown groups.
- Moving of F-Runtime groups that do not contain instances of F-Block types to another task.

12.8.2.3 Changes that require a cold restart or warm restart (restart) of the F-CPU

The following changes take effect only after a cold restart or warm restart of the F-CPU:

- Changes in values of the ID or R_ID parameter of the F-blocks F_SENDR/BO, F_RCVR/BO, F_SDS_BO and F_RDS_BO. (See also section " Change in the safety-related communication between F-CPU's (Page 208) ".)

Note

In the PCS 7 process control system and when blocks from PCS 7 libraries are used, the startup type "Cold restart" is not permitted.

12.8.2.4 Changes that require an F-CPU STOP in a single CPU

You can make exactly the same changes to the hardware configuration in an S7 FH System as in an S7 H System; see Manual " Automation System S7-400H Fault-tolerant Systems (<http://support.automation.siemens.com/WW/view/en/82478488>) ".

If you are operating a non-redundant F-CPU, an F-CPU STOP is required to download these changes.

Special features for S7 FH Systems:

- The F-I/O can receive modified parameters in an S7 FH System only after removal and insertion. The F-I/O detect a communication error after the first change is downloaded.

12.8.2.5 Changing the time ratios or F-Monitoring times

Make sure that the time monitoring functions are not triggered when the time ratios or F-Monitoring times are changed.

- Changing the OB cycle time

Procedure for changing the OB cycle time

1. Using the newly specified value for the OB cycle time, calculate the minimum F-Monitoring times for:
 - F-Cycle time monitoring at input MAX_CYC at the F_CYC_CP F-control block
 - TIMEOUT inputs of the F-Blocks for safety-related communication between F-CPU's
 - TIMEOUT inputs of the F-Blocks for data exchange between F-Shutdown groups
 - F-I/O

For more information about the F-monitoring time, refer to Chapter " Run times, F-Monitoring times, and response times (Page 460) ".

2. If the values assigned up to now are less than the newly calculated values, you must increase the F-Monitoring times prior to changing the OB cycle time. Compile the S7 program and download the changes.
3. Change the OB cycle time.

Note

Changing the OB cycle time involves a change in the hardware configuration. Refer to chapter " Changes that require an F-CPU STOP in a single CPU (Page 206) ".

- Moving of F-Runtime groups to a different task
Corresponds to a change of the OB cycle times of the relevant tasks (see above).
- Changing of F-Monitoring times for F-Blocks for safety-related communication between F-CPU's and for data exchange between F-Shutdown groups.
- Changing the F-Monitoring times of an F-I/O.

Note

Changing the F-Monitoring times of an F-I/O involves a change in the hardware configuration. Refer to chapter " Changes that require an F-CPU STOP in a single CPU (Page 206) ".

When changing these F-Monitoring times, ensure that the values do not fall below the calculated minimum F-Monitoring times. For more information about the F-monitoring time, refer to Chapter " Run times, F-Monitoring times, and response times (Page 460) ".

12.8.2.6 Change in the safety-related communication between F-CPU's

Introduction

If the safety-related communication between F-CPU's is to continue to run in all phases, you must proceed in multiple steps.

Rule

You must never simultaneously change the interconnection for a send data element at F_SENDBO/F_SDS_BO/F_SENDR and for the associated receive data element at F_RCVBO/F_RDS_BO/F_RCVR. The simultaneous activation of the new interconnections is otherwise not ensured.

Procedure for changing interconnections

For changing an interconnection to a send data element of the F_SENDBO/F_SDS_BO/F_SENDR F-Blocks or from a receive data element of the F_RCVBO/F_RDS_BO/F_RCVR F-Blocks, the following sequence must be adhered to:

1. Interconnect the new data element to be sent with a previously unused input SD_BO_xx/SD_R_xx of the F_SENDBO/F_SDS_BO/F_SENDR. Compile the S7 program and download the change.

Result: The new data element is now available at the corresponding RD_BO_xx/RD_R_xx output of F_RCVBO/F_RDS_BO/F_RCVR.

2. Now, interconnect the blocks again to the new RD_BO_xx/RD_R_xx output for further processing of the received signals. Compile the S7 program and download the change.

Result: Through this method, you ensure a consistent switchover to the new data path.

3. Delete the superfluous interconnection at F_SENDBO/F_SDS_BO/F_SENDR.
4. Compile the S7 program and download the change.

Procedure for replacing the communication partner

When a communication partner is replaced, the following sequence must be adhered to:

1. Configure the new S7 connection in *NetPro*. Download the connection data in RUN mode.
2. Place a new instance of F_SENDBO/F_SDS_BO/F_SENDR on the sender side. Assign the data for the new S7 connection to the ID and R_ID inputs. Interconnect the new data element to be sent with the SD_BO_xx/SD_R_xx inputs of the F_SENDBO/F_SDS_BO/F_SENDR. Compile the S7 program and download the change.
3. Place a new instance of F_RCVBO/F_RDS_BO/F_RCVR on the receiver side. Assign the data for the new S7 connection to the ID and R_ID inputs.

Compile the S7 program and download the change.

Result: The data of the old and new communication partner are now available to you on the receiver side.

4. Now, interconnect the blocks again to the RD_BO_xx/RD_R_xx outputs of the new R_RCVBO/F_RDS_BO/F_RCVR for further processing of the received signals.

Delete the superfluous F_RCVBO/F_RDS_BO/F_RCVR. Compile the S7 program and download the change.

Result: Through this method, you ensure a consistent switchover to the new communication partner.

5. Delete the superfluous F_SENDBO/F_SDS_BO/F_SENDR. Compile the S7 program and download the change.

6. If applicable, delete the superfluous connection from *NetPro*. Download the connection data in RUN mode.

12.8.2.7 Initial run and startup characteristics

Newly inserted F-Blocks execute an initial run after online changes. In this regard, note the startup characteristics described in the block descriptions. In cases where the initial run is not specifically mentioned, the behavior described after an F-Startup also applies to the initial run.

12.9 Deleting the safety program

Procedure

To delete a safety program from an F-CPU, follow these steps:

1. Delete all F-Charts from the chart folder. The symbols of these charts are highlighted with a yellow background in *SIMATIC Manager*.
2. Delete all charts whose name begins with "@F_".
3. Compile the S7 program with the "Generate module drivers" option selected.
4. In *HW Config*, open the properties dialog for the F-CPU from which you want to delete the safety program. Clear the "CPU contains safety program" option under "Protection".
5. Compile the hardware configuration.
6. Compile the S7 program.

12.10 Acceptance test following system upgrade

Acceptance after a system upgrade

The table shows whether a migration to *S7 F Systems V6.2* changes the signature and necessitates a STOP of the F-CPU or a new acceptance.

| Migration from | Change of signature | STOP of F-CPU required | New acceptance required |
|--|---------------------|------------------------|-------------------------|
| <i>S7 F Systems V6.0</i> (or higher) without update of the F-library (starting from <i>S7 F Systems Lib V1_3</i>) | No | No | No |
| <i>Failsafe Blocks (V1_2)</i> to <i>S7 F Systems Lib V1_3</i> | Yes | Yes | Changes |
| <i>Failsafe Blocks (V1_2) SPx</i> to <i>S7 F Systems Lib V1_3 SP1</i> | Yes | Yes | Changes |
| <i>S7 F Systems Lib V1_3</i> to <i>S7 F Systems Lib V1_3 SP1</i> | | | |
| • With use of the new F-blocks | Yes | No | Changes |
| • With use of the changed F_CH_DO | Yes | Yes ¹⁾ | Changes |
| • With use of the changed F_CH_BI | Yes | No | Changes |
| • With use of the changed F_QUITES | Yes | No | Changes |
| • With use of the changed F_CH_AI | Yes | No | Changes |
| • With use of the changed F_PA_AI | Yes | No | Changes |
| • With use of the changed F_SQRT | Yes | No | Changes |
| • With use of the changed F_CHG_BO | No | No | Changes |
| • With use of the changed F_CHG_R | No | No | Changes |
| <i>S7 F Systems Lib V1_3 SP1</i> and <i>SP2</i> | | | |
| • When the new F-blocks are used (F_SWC_CB, F_SWC_CR, F_CH_RI) | Yes | Yes | Changes |
| • With use of the changed F_XooY | Yes | Yes | Changes |
| • With use of the changed F_2oo3AI | Yes | Yes | Changes |
| • With use of the changed F_CH_AI | Yes | Yes | Changes |
| • With use of the changed F_TESTC / F_PLK | Yes | Yes | Changes |
| • With use of the changed F-blocks for F-communication: F_SDS_BO, F_SENDBO and F_SENDR | Yes | No | Changes |
| • With use of the changed F-blocks for F-communication: F_RDS_BO, F_RCVBO und F_RCVR | Yes | Yes | Changes |
| • When the changed F-block F_SWC_BO for Maintenance Override (MOS) is used | Yes | No | Changes |

1): The change is not safety-related and does not influence the usability of the existing project.

System Acceptance Test

13.1 Overview of system acceptance test

Introduction

During the system acceptance test, all relevant application-specific standards must be adhered to as well as the following procedures. This also applies to systems that are not subject to acceptance testing. For acceptance testing, you must note the systems requiring approval in the Certification Report.

As a general rule, the acceptance test of an F-System is performed by independent experts.

Special functions in *SIMATIC Manager* assist you for the acceptance test of an F-System. You can use these functions to:

- Compare safety programs
- Log safety programs
- Print safety programs

All data relevant to the acceptance test of the S7 F System can be archived in *SIMATIC Manager* (**File > Archive**) and printed as needed.

For more information, refer to Chapters "Comparing safety programs (Page 185)", ""Logs..." button (Page 182)" and "Printing project data of the safety program (Page 192)".

13.2 Commissioning a safety program

13.2.1 Initial acceptance test of a safety program

General procedure for the initial acceptance test of a safety program

1. Preliminary test of the configuration of the F-CPU and F-I/O (optional)
2. Backup of the *STEP 7* project
3. Inspection of the printout
4. Downloading the S7 program to the F-CPU
5. Implementation of a complete function test

13.2.2 Preliminary test of the configuration of the F-CPU and F-I/O (optional)

Introduction

After you finish configuring the hardware and assigning parameters for the F-CPU and F-I/O, you can perform an initial acceptance test for the F-I/O configuration.

In order to do this, the hardware configuration data must be printed out, checked, and saved together with the overall *STEP 7* project.

Printing hardware configuration data

1. Select the correct F-CPU or S7 program assigned to it.
2. In the *SIMATIC Manager* select the menu command **Options > Edit safety program**.
The "Safety Program" dialog will appear.
3. Click the "Print" button and select the "HW Configuration" option in the next dialog:
4. Select "All" for the print range, and select the "Module description" and "Address list" options there. In addition, select the "Including parameter description" option to include your parameter descriptions in the printout.

Checking hardware configuration data

1. Check the parameters of the F-CPU in the printout.

In safety mode, access by means of the F-CPU password must not be authorized when making changes to the standard user program, since changes to the safety program can also be made. To rule out this possibility, you must configure **Protection Level 1**. In addition, you must select the "CPU contains safety program" option. The corresponding protection level and "CPU contains safety program" is included in the printout.

2. Check the safety-related parameters of the F-I/O in the printout.

These safety-related parameters can be found in the printout for the respective F-I/O. The data are structured differently according to the F-I/O as follows:

SM 326; DI 24 x DC 24V (article no. 6ES7326-1BK00-0AB0), SM 326; DI 8 x Namur, SM 326 DO 10 x DC 24V/2A and SM 336; AI 6 x13 Bit

- The PROFIsafe source address does not appear in the printout.
- You determine the PROFIsafe destination address from the address value under "Addresses – Inputs – Start". Divide this address value by "8".
- The safety-related parameters are found under "Parameters – Basic Settings" or "Parameters – Input/Output x".

Fail-safe modules ET 200S, ET 200SP, ET 200pro, ET 200eco, ET 200iSP, SM 326; DI 24 x DC 24V (as of article no. 6ES7326-1BK01-0AB0) and SM 326; DO 8 x DC 24V/2A PM

- The PROFIsafe source address is found under "Parameters – F-Parameters – F_Source_Address".
- The PROFIsafe destination address is found under "Parameters – F-Parameters – F_destination_address".
- The safety-related parameters are found under "Parameters – F-Parameters" and "Parameters – Module parameters".

Fail-safe DP standard slaves/IO standard devices

- The PROFIsafe source address is found under "PROFIsafe – F_Source_Add".
- The PROFIsafe destination address is found under "PROFIsafe – F_Dest_Add".
- The safety-related parameters are found under "PROFIsafe".

For information on handling of any technological safety-related parameters, refer to the documentation for the respective DP standard slave/IO standard device.

3. Once the safety-related parameters of an F-I/O module are checked, the parameter CRCs in the printout are sufficient as reference for further acceptance testing. These parameter CRCs have the following appearance (address/F-address = PROFIsafe address):

Fail-safe signal modules S7-300 (SM 326; DI 24 x DC 24V, with article no. 6ES7326-1BK00-0AB0; SM 326; DI 8 x NAMUR; SM 326; DO 10 x DC 24V/2A; SM 336; AI 6 x 13-bit)

- Parameter CRC: 12345
- Parameter CRC (excluding F-addresses): 54321

Fail-safe modules ET200S, ET 200SP, ET 200pro, ET 200eco, ET 200iSP and S7-300 fail-safe signal modules (SM 326; DI 24 x DC 24 V, as of article no. 6ES7326-1BK01-0AB0; SM 326; DO 8 x DC 24V/2A PM)

- Parameter CRC: 12345
- Parameter CRC (excluding F-addresses): 54321


Fail-safe DP standard slaves/IO standard devices

- F_Par_CRC: 12345
- F_Par_CRC (excluding F-addresses): 54321

F-I/O that are to be assigned the same safety-related parameters can be copied during configuration. All safety-related parameters for these no longer have to be checked individually: It is sufficient to compare every other CRC (for example, "Parameter CRC (excluding address)") of the copied F-I/O to the corresponding CRC of the previously checked F-I/O and to check the PROFIsafe source and destination addresses.

4. Check that the PROFIsafe addresses are unique from one another.

To determine the PROFIsafe addresses of individual F-I/O, refer to step 1.

| |
|---|
|  WARNING |
| <p>Address assignment in subnets only and in mixed configurations</p> <p>The following applies to PROFIBUS DP subnets only:</p> <p>The PROFIsafe destination address and, thus, the switch setting on the address switch of the F-I/O must be unique network-wide* and station-wide** (system-wide). You can assign up to 1022 different PROFIsafe destination addresses.</p> <p>The following applies to PROFINET IO subnets only and to mixed configurations of PROFIBUS DP and PROFINET IO:</p> <p>The PROFIsafe destination address and, thus, the address switch setting on the F-I/O must be unique only*** within the PROFINET IO subnet, including all lower-level PROFIBUS DP subnets, and station-wide** (system-wide).</p> <p>For S7-300 F-SMs and ET 200S, ET 200eco, ET 200iSP and ET 200pro F-modules, you can assign a maximum of 1022 different PROFIsafe destination addresses.</p> <p>A PROFINET IO subnet is characterized by the fact that the IP addresses of all networked nodes have the same subnet address, i.e. the IP addresses match in the positions that have the value "1" in the subnet mask.</p> <p>Example:</p> <p>IP address: 140.80.0.2.</p> <p>Subnet mask: 255.255.0.0 = 11111111.11111111.00000000.00000000</p> <p>Meaning: Bytes 1 and 2 of the IP address define the subnet; subnet address: 140.80.</p> <p>* A network consists of one or more subnets. "Network-wide" means across subnet boundaries.</p> <p>** "Station-wide" means for one station in <i>HW Config</i> (e.g. an S7-400H station).</p> <p>*** Across Ethernet subnets, excluding cyclic PROFINET IO communication (RT communication)</p> |

13.2.3 Backup of the STEP 7 project

Requirements

Prior to the acceptance test, compile the safety program to be tested.

Backing up and archiving

The safety program that is to undergo approval testing must be backed up and archived with the complete *STEP 7* project. You must print out all of the project data *unfiltered* and archive them together with the *STEP 7* project:

- Chart (standard chart and F-Chart)
- Safety program: Block lists and signatures
- Safety-related parameters
- HW configuration
- Compilation log
- Download log

The procedure for backing up and archiving *STEP 7* projects is described in the basic help of *STEP 7*.

13.2.4 Inspection of the printout

Introduction

Print the entire project as described in the section "Printing project data of the safety program (Page 192)".

Printout

The printout contains the collective signature as a reference. The collective signature appears in the printout at two positions. All values must match the value in the footer.

- In the program information section as a value of the block container:
 - For the current compilation
 - For the reference
 - For the last online change (optional)
- In the footer as a value from the source

The following relations must be checked depending on an online change:

- If no online change has taken place, the collective signature for the current compilation must match the collective signature in the footer.
- If an online change has taken place, the collective signature in the footer corresponds to the collective signature of the last online change.

If a collective signature is not printed in the footer, this means that the safety program or the configuration (*HW Config* or *NetPro*) has changed. In this case, you must recompile the safety program.

The version number of the utilized *S7 F Systems* optional package appears in the footer of the printout and must be checked by you.

Check of safety-related parameters

Check the values of all safety-related parameters in the corresponding section of the printout for the safety program.

The following will be printed out:

- Values of all non-interconnected, invisible input parameters
- Values of all special input parameters to be checked, such as F-Monitoring times

A marking occurs in the printout:

- Marking "(*)":

Values of all output parameters for which the runtime sequence does not correspond to the data flow

This is the case if the F block is first called after the output parameter was already transferred to another F block, for example, in a feedback loop.

- Marking "(!)":

Inputs or outputs on an F block that have been identified by the system as parameters to be taken into account in the printout

Checking the signatures and initial value signatures of the F-blocks

The signatures and initial value signatures of all F-blocks must match those in Annex 1 of the Certificate Report.

Checking the signatures and initial value signatures of the F-block types

The signatures and initial value signatures of all F-block types must match those in the acceptance test documents of the F-block types (see section "Acceptance test of F-Block types (Page 222)").

The acceptance test documents of the F-block types also list the signatures and initial value signatures of all called F-blocks. These signatures must also match those in the safety program.

13.2.5 Downloading the S7 program to the F-CPU

Introduction

Download the S7 program to the F-CPU as described in section "Downloading the safety program (Page 198)". Then check the signatures.

Checking the collective signatures

After downloading the S7-program to the F-CPU you have to compare the collective signature of the safety program in the F-CPU with the collective signature in the accepted printout. S7 FH Systems must be in redundant system state and safety mode must be activated.

You can get the collective signature of the safety program and the signatures of the F-blocks in the F-CPU with the menu command **Options > Edit safety program**.

13.2.6 Implementation of a complete function test

Overview

Requirements

For successful initial acceptance of a safety program, a complete function test is required.

For this purpose, corresponding test specifications must be implemented based on documented procedures in order to verify the configured safety functions and rule out unwanted side effects.

The following points must be observed:

- Conformity to the specification of the safety function
- Full coverage of the safety program during the function test

Note

The system charts created by S7 F Systems with prefix "@F_" do not have to be tested.

- Negative tests
- Tests for the time sequence and logic sequence

Results

The results of the function test must be documented, The following information should be present:

- Collective signature of the safety program
- Safety program printout
- Any utilized test tools including version
- Name of responsible persons

- Test description
- Test result

13.3 Acceptance test of safety program changes

Procedure

To perform an acceptance test on your safety program changes, follow these steps:

1. Back up your safety program.
2. Compare your new safety program with your accepted safety program. For more information, refer to Chapter "Comparing safety programs (Page 185)".
3. Inspect the changes in the printout. You must locate the changes that you made to your safety program on the printout again. Check the signature in the printout (and in the footer). To do so, follow the same procedure as for the initial acceptance test.
4. Download your modified safety program to the F-CPU.
5. Perform a function test of your changes.

13.4 Acceptance test of F-Block types

Initial acceptance

The same process is used for initial acceptance of a newly created F-block type as for initial acceptance of a safety program. The function test of the F-block type must take place in a different safety program than the test environment.

For acceptance of F-block types, the signature and initial value signature of the resulting generated F-block are relevant. You can obtain these signatures from the printout of the safety program. In addition, you must also check the signatures and initial value signatures of the called F-blocks.

The collective signatures in the footers of the printouts of the safety program and the CFC chart of the F-block type must match. Otherwise, you must recompile the F-block type.

All F-blocks called in an F-block type must be compared.

Note

For testing a safety program in which an F-block type is used, you must check the signatures of the F-block type and the signatures of all called F-blocks.

Acceptance of changes

The process for acceptance of changes to an F-block type is the same as for a safety program.

For acceptance of the F-block types, use a printout to document the signature and initial value signature of the new F-block type as well as the signatures and initial value signatures of all F-blocks called in the F-block type.

In addition, you must use a function test to test all points in the test safety program at which the new F-block type is called. Changed signatures of F-blocks are displayed in the chart view when safety programs are compared.

Operation and Maintenance

14.1 Notes on safety mode of the safety program

Introduction

The rules and safety information for operation of S7 F/FH Systems is presented below.

Using simulation devices / simulation programs

WARNING

If you operate simulation devices or simulation programs that generate safety message frames, e.g., in accordance with PROFIsafe, and make them available to the S7 F/FH System via the bus system (e.g., PROFIBUS DP), you must ensure the safety of the F-system using organizational measures, e.g., such as operational monitoring and manual safety shutdown.

If you use the *S7-PLCSIM* function of *STEP 7* to simulate safety programs, these measures are not necessary because *S7-PLCSIM* cannot establish an online connection to a real S7 component.

Note, for example, that a protocol analyzer may not perform functions that reproduce recorded message frame sequences with correct time behavior.

STOP by means of ES operation, mode selector, or communication function

WARNING

Switching from STOP to RUN mode by means of an ES operation, mode selector, or communication function is not interlocked. For example, only one keystroke on the ES is necessary to switch from STOP to RUN mode. For this reason, a STOP that you have set by means of an ES operation, mode selector, or communication function must not be regarded as a safety condition.

Therefore, always switch off the F-CPU directly at the device when performing maintenance work.

Placing F-CPU in STOP with SFC 46 "STP"

 **WARNING**

A STOP state initiated with SFC 46 "STP" can be canceled very easily (and unintentionally) by means of an ES operation. For this reason, an F-CPU STOP initiated by SFC 46 is not a fail-safe STOP.

Fiber-optic cable between the synchronization modules in S7 F/FH Systems

 **WARNING**

Two F-CPU not simultaneously as master system

In S7 F/FH Systems, you must ensure that the two F-CPU are not master systems simultaneously. Otherwise, this could lead to dangerous errors.

This situation (both F-CPU as master simultaneously) can occur if the two fiber-optic cables used to connect the F-CPU in S7 F/FH Systems in the redundant system state are unplugged or interrupted simultaneously. You must prevent this by routing the fiber-optic cables separately.

This situation (both F-CPU as master simultaneously) can also occur after an F-CPU is repaired if the F-CPU have not yet been connected using *both* fiber-optic cables prior to switching on the power supply.

You must implement organizational measures to ensure following replacement of an F-CPU that both connections are established using the fiber optic cables *prior* to switching on the power supply.

Additional Information

Information about replacing components in fault-tolerant systems can be found in Manual "Automation System S7-400H Fault-tolerant Systems (<http://support.automation.siemens.com/WW/view/en/82478488>)".

14.2 Replacing software and hardware components

Replacement of software components

When you replace software components on your ES, e.g. in case of new versions of *PCS 7* or *STEP 7*, you must observe the information on upward and downward compatibility in the documentation and in the readme files of these products.

Installing new versions of the software packages

After installation of a new version of *PCS 7*, *STEP 7* or the optional packages *CFC*, *SCL*, etc., follow these steps:

1. Compile the S7 program in the new environment.
2. Compare the collective signature of the newly compiled S7 program with the collective signature of the accepted safety program (see also "Checking the collective signature" in section "Commissioning a safety program (Page 214)").
3. If the collective signatures are identical, the safety programs match.
4. If the collective signatures are not identical, the safety program has changed. In this case, follow the same procedure as for a change of the safety program.

Replacement of hardware components

You replace hardware components for S7 F/FH Systems (modules, batteries, etc.) the same as in standard mode.

Removal and insertion of F-I/O during operation

F-I/O can be removed and inserted during operation in exactly the same way as standard I/O. Note however that the replacement of an F-I/O during operation may trigger a communication error in the F-CPU.

You must acknowledge the communication error at the ACK_REI input of the F-channel driver in your safety program. Without acknowledgment, the F-I/O remains passivated.

CPU operating system update

Check of the CPU operating system for F-validity: When a new CPU operating system is used (operating system update), you must check whether the utilized CPU operating system is permitted for use in an F-system.

The annex of the certificate specifies the minimum CPU operating system version that ensures fail-safe compatibility. This specification and any information about the new CPU operating system must be observed.

Operating system update for interface modules

When a new operating system is used for an interface module, e.g. IM 151-1 HIGH FEATURE ET 200S (operating system update, see online help *STEP 7*), you must observe the following:

If you have selected the "Activate firmware after download" check box for the operating system update, the IM is automatically reset after a successful download and then runs with the new operating system. All F-I/O are passivated after startup of the IM.

The reintegration of the F-I/O is performed in the same way as after a communication error, i.e. by an acknowledgement at the ACK_REI input of the F-channel driver.

Duration of repair of S7 F/FH Systems

For S7 F/FH Systems, the repair for redundant components should be organized in such a way that the duration of the repair following a failure does not exceed 24 hours if possible. At unmanned plants, a repair duration of 72 hours is permitted on weekends. In general, availability increases as the repair duration decreases.

Fiber-optic cables with S7 F/FH Systems

After repair of an F-CPU, you must not withdraw fiber-optic cables simultaneously from the F-CPU.

Preventive maintenance (Proof Test)

For an ordinary configuration, the probability values for the certified components of the F-system guarantee a service life (proof-test interval) of 20 years.

For detailed information, refer to the F-I/O manuals. Proof test for complex electronic components usually means replacement with unused goods.

A shorter proof-test interval is usually required for sensors and actuators.

Uninstallation of S7 F Systems

For information on uninstallation of software, refer to section "Installing the S7 F Systems optional package (Page 29)".

You disassemble and dispose of the hardware of an F-system in the same way as for standard automation systems. For more information, refer to the *hardware manuals*.

14.3 F-Forcing

Introduction

Depending on the CFC version you are using, *S7 F Systems* V6.1 and higher with *S7 F Systems Lib V1_3* SP1 and higher supports the forcing of F-parameters in deactivated safety mode.

F-Forcing allows you to modify F-Parameters at user interconnections.

- The modification of F-Parameters at system interconnections is not supported.
- Changing force values with activated F-forcing is not supported for F-parameters.

Consult the documentation for CFC or PCS 7 to find out which CFC versions support forcing of F-Parameters, in particular.



WARNING

Using the "F-Forcing" function

Forcing is only permitted when the safety of the system is ensured by other measures.

Procedure

1. Configure forcing for F-Parameters in CFC using the same procedure as for forcing with standard parameters.
2. If you haven't already done so, you will be prompted to deactivate safety mode.
 - Modify and check the force values for F-Parameters.
 - Enable F-Forcing for F-Parameters.
3. In your CFC program, make changes to F-Parameters of user interconnections by means of F-Forcing.
4. Activate safety mode again when forcing is no longer taking place in the F-Parameters.

Note

F-Forcing is deactivated automatically any time the F-Program starts up. The display in the CFC Editor is not updated after startup, however. The display can be updated by deactivating/activating safety mode again, for example.

The F-Program starts up:

- Each time the CPU restarts (cold/warm restart), e.g., following a brief power outage
- Each time the CPU restarts after a full shutdown

Note

Safety mode cannot be activated if F-Forcing is activated for an F-Parameter.

Note

F-Forcing is a typical commissioning function. The final F-Program should not include F-Forcing of F-Parameters.

Use the Maintenance Override function for the maintenance functions.

See also

Operator inputs with the "Secure Write Command++" function (Page 123)

F-libraries

A.1 Overview of the S7 F Systems Lib V1_3 SP2 F-library

A.1.1 F-Blocks

Overview

The *S7 F Systems Lib V1_3 SP2* F-library contains the following:

- In the block container F-User Blocks\Blocks: F-blocks
- In the block container F-Control Blocks\Blocks: F-control blocks

Note

Refer also to sections "Differences between the F-Libraries Failsafe Blocks (V1_x) and S7 F Systems Lib V1_3 (Page 434)" and "Differences between the S7 F Systems Lib F-libraries (Page 456)".

Note

You are not permitted to change the name of the F-library.

Note

FB numbers of F-blocks

You are not permitted to change the numbers of the F-blocks.

The following F-blocks present in the *S7 F Systems Lib V1_3 SP2* use FBs that are also used in *S7 Distributed Safety*:

| <i>S7 F Systems Lib V1_3 SP2</i> | Number of the FB | F-library <i>Distributed Safety (V1)</i> |
|----------------------------------|------------------|--|
| F_CH_DII | FB 465 | F_IGNTR |
| F_CH_DIO | FB 466 | F_TIGHTN |
| F_POLYG | FB 467 | F_GAS_BU |
| F_INT_P | FB 468 | F_OIL_BU |
| F_PT1_P | FB 469 | F_AIRD |

A.1.2 F-Data types


Function

Special F-Data types in a safety data format are used for fail-safe block interfaces. The safety data format is used to expose data and address errors.

Example

```
F_BOOL:
                                     STRUCT
DATA                                BOOL
PAR_ID                               WORD
COMPLEM                              WORD
                                     END_STRUCT
```

If you want to change the value (default) of a block interface with an F-Data type, you can only change the DATA component.

| |
|--|
|  WARNING |
| Values of PAR_ID and COMPLEM must not be changed |
| You must not change the PAR_ID and COMPLEM components after the S7 program has been compiled since this might result in serious errors remaining undetected. If errors in the safety data format are detected while the safety program is running, an F-STOP is triggered. You must recompile the S7 program and download it to the F-CPU, if necessary. |

A.1.3 Block interfaces

Note the following special features regarding the block interfaces of F-Blocks:

- The EN and ENO inputs/outputs are neither evaluated nor assigned by the program code of the F-Block and you must not interconnect them.
- All F-Blocks have additional inputs/outputs in addition to the inputs/outputs documented in the following block descriptions. These are initialized or interconnected automatically when the S7 program is compiled and you must not change them. Online changes affecting non-documented inputs/outputs can lead to an F-STOP. You can overcome manipulations to these inputs/outputs by recompiling the S7 program.

A.1.4 Behavior of F-Blocks with floating-point operations in the event of a number range overflow

The "Overflow (\pm infinity)", "Denormalized floating-point number", or "Invalid floating-point number (NaN)" events are:

- Either output at the output and available for further processed by the subsequent F-Blocks

or

- Signaled at special outputs. If necessary, a fail-safe value is output.

If the floating-point operation yields an invalid floating-point number (NaN) and an invalid floating-point number (NaN) does not already exist as an address, the following diagnostic event is entered in the diagnostic buffer of the F-CPU:

- "Safety program: invalid REAL number in DB" (Event ID 16#75D9)

You can use this diagnostic buffer entry to identify the F-Block with the invalid floating-point number (NaN).

Refer also to the documentation of the F-Blocks.

If you cannot rule out the occurrence of these events in your safety program, you must decide independently of your application whether you have to react to these events in your safety program. With F-Block F_LIM_R, you can check the result of a floating-point operation for overflow (\pm infinity) and invalid floating-point number (NaN).

A.1.5 Behavior of F-Blocks in the event of safety-related faults

If F-Blocks or F-Control blocks detect a safety-related fault, they trigger a fault reaction. Error information is entered in the diagnostic buffer of the F-CPU. The online help for the diagnostic events provides detailed information and suggests corrective actions.

The respective fault reactions and other diagnostic options can be found in the documentation for the F-Blocks and F-Control blocks.

A.2 S7 F Systems Lib V1_3 SP2 F-blocks

A.2.1 Logic blocks with the BOOL data type

A.2.1.1 Logic Blocks of the BOOL Data Type

Overview

| Block name | Block number | Description |
|------------|--------------|---|
| F_AND4 | FB 301 | AND logic operation on four inputs |
| F_OR4 | FB 302 | OR logic operation on four inputs |
| F_XOR2 | FB 303 | XOR logic operation on two inputs |
| F_NOT | FB 304 | NOT logic operation |
| F_2OUT3 | FB 305 | 2oo3 evaluation of inputs of data type BOOL |
| F_XOUTY | FB 306 | XooY evaluation of inputs of data type BOOL |

A.2.1.2 F_AND4: AND logic operation on four inputs

Function

This block links the INx inputs by means of AND. The OUT output is "1" when all INx inputs are "1". Otherwise the OUT output is "0". The OUTN output corresponds to the negated OUT output.

Truth table

| IN1 | IN2 | IN3 | IN4 | OUT | OUTN |
|-----|-----|-----|-----|-----|------|
| 0 | 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 0 | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 | 1 |
| 0 | 1 | 0 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 0 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 | 1 |
| 1 | 1 | 0 | 0 | 0 | 1 |

| IN1 | IN2 | IN3 | IN4 | OUT | OUTN |
|-----|-----|-----|-----|-----|------|
| 1 | 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 | 0 |

Inputs/outputs

| | Name | Data type | Description | Default |
|-----------------|------|-----------|----------------|---------|
| Inputs: | IN1 | F_BOOL | Input 1 | 1 |
| | IN2 | F_BOOL | Input 2 | 1 |
| | IN3 | F_BOOL | Input 3 | 1 |
| | IN4 | F_BOOL | Input 4 | 1 |
| Outputs: | OUT | F_BOOL | Output | 1 |
| | OUTN | F_BOOL | Negated output | 0 |

Error handling

None

A.2.1.3 F_OR4: OR logic operation on four inputs

Function

This F-Block combines the INx inputs with a logical OR. The OUT output is "1" when at least one INx input is "1". If all INx inputs are "0", the OUT output is "0". The OUTN output corresponds to the negated OUT output.

Truth table

| IN1 | IN2 | IN3 | IN4 | OUT | OUTN |
|-----|-----|-----|-----|-----|------|
| 0 | 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 0 | 1 | 1 | 0 |
| 0 | 0 | 1 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 | 0 |
| 1 | 0 | 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 | 0 |

| IN1 | IN2 | IN3 | IN4 | OUT | OUTN |
|-----|-----|-----|-----|-----|------|
| 1 | 0 | 1 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 1 | 1 | 0 |

Inputs/outputs

| | Name | Data type | Description | Default |
|-----------------|------|-----------|----------------|---------|
| Inputs: | IN1 | F_BOOL | Input 1 | 0 |
| | IN2 | F_BOOL | Input 2 | 0 |
| | IN3 | F_BOOL | Input 3 | 0 |
| | IN4 | F_BOOL | Input 4 | 0 |
| Outputs: | OUT | F_BOOL | Output | 0 |
| | OUTN | F_BOOL | Negated output | 1 |

Error handling

None

A.2.1.4 F_XOR2: XOR logic operation on two inputs

Function

This F-Block combines the INx inputs with an exclusive OR. The OUT output is "1" if exactly one INx input is "1". The OUTN output corresponds to the negated OUT output.

Truth table

| IN1 | IN2 | OUT | OUTN |
|-----|-----|-----|------|
| 0 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |

Inputs/outputs

| | Name | Data type | Description | Default |
|----------------|------|-----------|-------------|---------|
| Inputs: | IN1 | F_BOOL | Input 1 | 0 |
| | IN2 | F_BOOL | Input 2 | 0 |

| | Name | Data type | Description | Default |
|----------|------|-----------|----------------|---------|
| Outputs: | OUT | F_BOOL | Output | 0 |
| | OUTN | F_BOOL | Negated output | 1 |

Error handling

None

A.2.1.5 F_NOT: NOT logic operation

Function

This F-Block inverts the input.

Truth table

| IN | OUT |
|----|-----|
| 0 | 1 |
| 1 | 0 |

Inputs/outputs

| | Name | Data type | Description | Default |
|---------|------|-----------|-------------|---------|
| Input: | IN | F_BOOL | Input | 0 |
| Output: | OUT | F_BOOL | Output | 1 |

Error handling

None

A.2.1.6 F_2OUT3: 2oo3 evaluation of inputs of data type BOOL

Function

This F-Block monitors three binary inputs for signal state "1". The OUT output is "1" when at least two INx inputs are "1". Otherwise the OUT output is "0". The OUTN output corresponds to the negated OUT output.

Truth table

| IN1 | IN2 | IN3 | OUT | OUTN |
|-----|-----|-----|-----|------|
| 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 | 1 |
| 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 1 | 0 |

Inputs/outputs

| | Name | Data type | Description | Default |
|-----------------|------|-----------|----------------|---------|
| Inputs: | IN1 | F_BOOL | Input 1 | 0 |
| | IN2 | F_BOOL | Input 2 | 0 |
| | IN3 | F_BOOL | Input 3 | 0 |
| Outputs: | OUT | F_BOOL | Output | 0 |
| | OUTN | F_BOOL | Negated output | 1 |

Error handling

None

A.2.1.7 F_XOUTY: XooY evaluation of inputs of data type BOOL

Function

The F-block monitors up to 16 binary inputs IN1...IN16 for signal state 1. The input signals are monitored for signal state 1 beginning with input IN1 up to an including input INY. The number of binary inputs to be monitored is set with the Y parameter. The OUT output is 1, when at least x inputs IN1...IN16 are 1. Otherwise, output OUT is 0. The OUTN output corresponds to the negated OUT output.

The binary inputs must be assigned consecutively beginning from IN1. When $X > Y$, $X \leq 0$, $X > 16$, $Y \leq 0$, then output OUT is 0. When $Y > 16$, the OUT output behaves the same as when $Y = 16$.

The OUT_XA output gives the number of active inputs, enabling larger functions such as "5oo32" with a significantly reduced block count.

I/Os

| | Name | Data type | Explanation | Default |
|-----------------|--------|-----------|--|---------|
| Inputs: | IN1 | F_BOOL | Input 1 | 0 |
| | IN2 | F_BOOL | Input 2 | 0 |
| | IN3 | F_BOOL | Input 3 | 0 |
| | ... | | ... | |
| | IN16 | F_BOOL | Input 16 | 0 |
| | X | F_INT | Minimum number of inputs with signal state 1: $0 < X \leq 16$ | 0 |
| | Y | F_INT | Number of inputs to be monitored: $0 < Y \leq 16$ | 0 |
| Outputs: | | | | |
| | OUT | F_BOOL | Output | 0 |
| | OUTN | F_BOOL | Negated output | 1 |
| | OUT_XA | F_REAL | Number of inputs with signal state 1 | 0 |

Error handling

An F-STOP is triggered when there is an error in the safety data format in the associated instance DB. The following diagnostics event is then entered in the diagnostics buffer of the F-CPU:

- "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.2 F-Blocks for F-Communication between F-CPU's

A.2.2.1 F-Blocks for F-Communication between F-CPU's

Overview

| F-Block name | Block number | Description |
|--------------|--------------|--|
| F_SENDBO | FB 370 | Sending of 20 data elements of data type F_BOOL in a fail-safe manner to another F-CPU |
| F_RCVBO | FB 371 | Receiving of 20 data elements of data type F_BOOL in a fail-safe manner from another F-CPU |
| F_SENDR | FB 372 | Sending of 20 data elements of data type F_REAL in a fail-safe manner to another F-CPU |
| F_RCVR | FB 373 | Receiving of 20 data elements of data type F_REAL in a fail-safe manner from another F-CPU |

| F-Block name | Block number | Description |
|--------------|--------------|--|
| F_SDS_BO | FB 352 | Sending of 32 data elements of data type F_BOOL in a fail-safe manner to another F-CPU |
| F_RDS_BO | FB 353 | Receiving of 32 data elements of data type F_BOOL in a fail-safe manner from another F-CPU |

A.2.2.2 F_SENDBO: Sending of 20 data elements of data type F_BOOL in a fail-safe manner to another F-CPU


Function

The F-block F_SENDBO sends the data of data type F_BOOL at the SD_BO_xx inputs in a fail-safe manner to another F-CPU. The data must be received there with the F-block F_RCVBO.

At the EN_SEND input, you can temporarily switch off communication between the F-CPU's in order to reduce the bus load by supplying the EN_SEND input with 0 (default setting = 1). Send data are then no longer sent to the associated F_RCVBO, and F_RCVBO provides the assigned fail-safe values for this time period. If communication was already established between the connection partners, a communication error is detected.

At the ID input, you must specify the local ID – from the perspective of the F-CPU – of the S7 connection (from connection table in *NetPro*).

Communication between the F-CPU's is implemented hidden in the background using a special safety protocol. For this purpose, you must define the communication relationship between an F_SENDBO in an F-CPU and an F_RCVBO in the other F-CPU by specifying an odd number at the R_ID input of F_SENDBO and F_RCVBO. Associated F_SENDBO and F_RCVBO are given the same value for R_ID.

| |
|---|
|  WARNING |
| <p>Value for the relevance address reference</p> <p>The value for the respective address relationship (input parameter R_ID; data type: DWORD) can be selected by the user but must be an odd number and unique network-wide for all safety-related communication connections. The value R_ID + 1 is internally assigned and must not be used. You must supply the ID and R_ID inputs with constant values when the F-block is called.</p> |

You must assign the desired F-monitoring time at the TIMEOUT input. The TIMEOUT input cannot be interconnected.

WARNING

Detecting and transmitting a signal level

It can only be ensured (from a fail-safe standpoint) that a signal level to be transmitted will be detected on the sender side and transmitted to the receiver if the signal level is present for at least as long as the assigned F-monitoring time (TIMEOUT).

For information regarding calculation of the F-monitoring time, see section " Run times, F-Monitoring times, and response times (Page 460) ".

Note

If the data are received with the F-block F_RCVBO of the *Failsafe Blocks* (V1_2) or (V1_1) F-library, you must assign the value "0" to the EN_SMODE input (default value = 1), because F_RCVBO will otherwise detect a CRC error.

Otherwise, you must leave the default value of the EN_SMODE input unchanged, because the operating mode of the F-CPU can otherwise not be evaluated with F_SENDBO at the SENDMODE output of the F-CPU.

I/Os

| | Name | Data type | Explanation | Default |
|-----------------|----------|-----------|---------------------------------------|---|
| Inputs: | EN_SEND | BOOL | 1 = Enable sending | 1 |
| | ID | WORD | Addressing parameter ID | W#16#0 |
| | R_ID | DWORD | Addressing parameter R_ID | DW#16#0 |
| | SD_BO_00 | F_BOOL | Send data element 00 | 0 |
| | ... | | ... | |
| | SD_BO_19 | F_BOOL | Send data element 19 | 0 |
| | CRC_IMP | DWORD | Address relationship CRC | DW#16#0 to be automatically supplied * |
| | TIMEOUT | F_TIME | F-monitoring time in ms | T#0ms |
| | EN_SMODE | F_BOOL | 1 = SENDMODE | 1 |
| Outputs: | ERROR | F_BOOL | 1 = Communication error | 0 |
| | SUBS_ON | F_BOOL | 1 = Receiver outputs fail-safe values | 0 |
| | RETVAl | WORD | Error code | W#16#0 |

*) The CRC_IMP input is automatically supplied when the S7 program is compiled and must not be changed. The CRC_IMP input is displayed as changed during the comparison of safety programs if changes were made to the connection configuration in *NetPro*.

Fail-safe value

Fail-safe values are output by the receiver F_RCVBO in the following cases:

- A communication error (e.g. CRC error, Timeout) was detected.
- The communication was disabled using EN_SEND = 0.
- An F-startup is present.

The SUBS_ON output is set to 1.

If the output of the fail-safe value is caused by a communication error, output ERROR = 1 is additionally set.

A "Timeout" communication error is detected for the first time when the communication between the F_SENDBO and F_RCVBO connection partners has already been established once. If communication cannot be established after startup of the sending F-system and receiving F-system, check the configuration of the safety-related CPU-CPU communication, the parameter assignment of the F_SENDBO and F_RCVBO and the bus connection. You can also obtain information on possible error causes by evaluating the RETVAL outputs of F_SENDBO and F_RCVBO. In general, always evaluate RETVAL of F_SENDBO and F_RCVBO as it may be that only one of the two outputs contains error information.

Reintegration

After a communication error, the data at the SD_BO_xx inputs are only output again by the receiver when no communication error is detected anymore and acknowledgement is made with a positive edge at the ACK_REI input of F_RCVBO.

Startup behavior

After start-up of the sending and receiving F-systems, the communication between the connection partners F_SENDBO and F_RCVBO must be established for the first time. The receiver F_RCVBO provides fail-safe values during the time period. The SUBS_ON output is set to 1.

RETVAl output

Non-fail-safe information about the type of communication error that occurred is provided at the RETVAL output for service purposes. You can read out this information via your ES/OS or evaluate it in your standard user program, if necessary. The DIAG bits are saved until an acknowledgement is made at the ACK_REI input of the associated F_RCVBO.

Structure of RETVAL

| Bit no. | Assignment | Possible error causes | Corrective actions |
|---------|-----------------------------------|--|---|
| Bit 0 | Reserve | — | — |
| Bit 1 | Receiver outputs fail-safe values | See bits 2-7 | Check bits 2-7 |
| Bit 2 | ERROR bit of USEND set | Basic communication problems of the internally called SFB 8 "USEND" detected | Bit 8-15 = Evaluate "STATUS" of SFB 8 "USEND" or SFB 9 "URCV" |

| Bit no. | Assignment | Possible error causes | Corrective actions |
|------------|---|--|--|
| | | See also description for bit 7 | See also description for bit 7 |
| Bit 3 | ERROR bit of USEND set | Basic communication problems of the internally called SFB 8 "USEND" detected | Bit 8-15 = Evaluate "STATUS" of SFB 8 "USEND" or SFB 9 "URCV" |
| | | See also description for bit 7 | See also description for bit 7 |
| Bit 4 | ERROR bit of URCV set | Basic communication problems of the internally called SFB 9 "URCV" detected | Bit 8-15 = Evaluate "STATUS" of SFB 8 "USEND" or SFB 9 "URCV" |
| | | See also description for bit 7 | See also description for bit 7 |
| Bit 5 | CRC error detected | See description for bit 7 | See description for bit 7 |
| Bit 6 | Sequence number error detected | See description for bit 7 | See description for bit 7 |
| Bit 7 | Timeout detected | Connection configuration not correct | Check and reload connection configuration |
| | | The bus connection to the partner F-CPU is faulty. | Check bus connection and ensure that no external sources of disturbance are present. |
| | | F-monitoring time of the F-CPU and of the partner F-CPU is set too low | Check the assigned F-monitoring time TIMEOUT for F_SENDBO and F_RCVBO of both F-CPU. Set higher value, if necessary. Recompile S7 programs and download them to the F-CPU. |
| | | STOP or internal fault of the CP | Switch the CPs to RUN. Check diagnostics buffer of the CPs. Replace the CPs, if necessary. |
| | | STOP, partial shutdown or full shutdown or internal fault of the F-CPU/partner F-CPU | Switch F-CPU to RUN. Perform F-startup. Check diagnostics buffer of F-CPU. Replace F-CPU, if necessary |
| | | Communication was disabled with EN_SEND = 0. | Enable communication again at the associated F_SENDBO with EN_SEND = 1 |
| | | S7 connection has changed, e.g. IP address of the CP was changed | Recompile S7 programs and download them to the F-CPU |
| Bit 8 - 15 | = "STATUS" error information of the internally called SFB 8 "USEND" or SFB 9 "URCV" | See description of the "STATUS" error information in the online help for SFB 8/SFB 9 or in the " System Software for S7-300/400 System and Standard Functions (http://support.automation.siemens.com/WW/view/en/1214574) " manual | — |

Error handling

An F-STOP is triggered when there is an error in the safety data format in the associated instance DB. The following diagnostics event is then entered in the diagnostics buffer of the F-CPU:

- "Safety program: Error in safety data format in DB" (Event ID 16#75DA)


A.2.2.3 F_RCVBO: Receiving of 20 data elements of data type F_BOOL in a fail-safe manner from another F-CPU

Function


The F-block F_RCVBO receives 20 data elements of data type F_BOOL from another F-CPU and makes it available to the RD_BO_xx outputs. The data must be sent from the other F-CPU with the F-block F_SENDBO.

At the ID input you must specify – from the perspective of the F-CPU – the local ID of the S7 connection (from the connection table in *NetPro*).

Communication between the F-CPU is implemented hidden in the background using a special safety protocol. For this purpose, you must define the communication relationship between an F_RCVBO in one F-CPU and an F_SENDBO in the other F-CPU by specifying an odd number at the R_ID input of F_SENDDP and F_RCVDP. Associated F_SENDBO and F_RCVBO receive the same value for R_ID.

| |
|--|
|  WARNING |
| Value for the respective address relationship |
| The value for the respective address relationship (input parameter R_ID; data type: DWORD) can be selected by the user but must be an odd number and unique network-wide for all safety-related communication connections. The value R_ID + 1 is internally assigned and must not be used. You must supply the ID and R_ID inputs with constant values when the F-block is called. |

You must assign the desired F-monitoring time at the TIMEOUT input. The TIMEOUT input cannot be interconnected.

| |
|---|
|  WARNING |
| Detecting and transmitting the signal level |
| It can only be ensured (from a fail-safe standpoint) that a signal level to be transmitted will be detected on the sender side and transmitted to the receiver if the signal level is present for at least as long as the assigned F-monitoring time (TIMEOUT). |
| For information regarding the calculation of the F-monitoring time, see section " Run times, F-Monitoring times, and response times (Page 460) ". |

The operating mode of the F-CPU with F_SENDDP is provided at the SENDMODE output. If the F-CPU with F_SENDDP is in deactivated safety mode, the SENDMODE output becomes = 1.

Note

If the data is received from an F_SENDBO block from older F-libraries, you must assign the COMMVER_USED input with 0. Otherwise, sequence number errors may occur. Default setting is "1".

I/Os

| | Name | Data type | Explanation | Default |
|-----------------|----------|---|--|---------------------------------------|
| Inputs: | ID | WORD | Addressing parameter ID | W#16#0 |
| | R_ID | DWORD | Addressing parameter R_ID | DW#16#0 |
| | CRC_IMP | DWORD | Address relationship CRC | DW#16#0 Is supplied automatically* |
| | TIMEOUT | F_TIME | F-monitoring time in ms | T#0ms |
| | ACK_REI | F_BOOL | Acknowledgment for reintegration | 0 |
| | SUBBO_00 | F_BOOL | Fail-safe value for receive data element 00 | 0 |
| | ... | | ... | |
| | SUBBO_19 | F_BOOL | Fail-safe value for receive data element 19 | 0 |
| COMMVER_USED | INT | 0 = Communication with blocks from older F-libraries < V1.3 SP2 (compatibility mode) 1 = Communication with blocks of F-Library V1.3 SP2 (or higher) | 1 | |
| Outputs: | ACK_REQ | BOOL | Acknowledgment for reintegration is required | 0 |
| | ERROR | F_BOOL | Communication error | 0 |
| | SUBS_ON | F_BOOL | Fail-safe values are output | 0 |
| | RD_BO_00 | F_BOOL | Receive data element 00 | 0 |
| | ... | | ... | |
| | RD_BO_19 | F_BOOL | Receive data element 19 | 0 |
| | SENDMODE | F_BOOL | 1 = F-CPU with F_SENDBO in deactivated safety mode | 0 |
| | RETVL | WORD | Error code | W#16#0 |

*) The CRC_IMP input is automatically supplied when the S7 program is compiled and must not be changed. The CRC_IMP input is indicated as changed during comparison of safety programs if changes were made to the connection configuration in *NetPro*.

Fail-safe values

The fail-safe values active at the SUBBO_xx inputs are output in the following cases:

- A communication error (e.g. CRC error, timeout) was detected.
- The communication was disabled at the associated F_SENDBO via EN_SEND = 0.
- An F-startup is present.

The SUBS_ON output is set to 1.


If the output of the fail-safe value is caused by a communication error, output ERROR = 1 is additionally set.

A "Timeout" communication error is detected for the first time when the communication between the F_SENDBO and F_RCVBO connection partners has already been established once. If communication cannot be established after startup of the sending and receiving F-system, check the configuration of the safety-related CPU-CPU communication, the parameter assignment of F_SENDDP and F_RCVDP and the bus connection. You can also obtain information on possible error causes by evaluating the RETVAL outputs of F_SENDBO and F_RCVBO. In general, always evaluate RETVAL of F_SENDBO and F_RCVBO as it may be that only one of the two outputs contains error information.

Reintegration

After a communication error, the data active at the SD_BO_xx inputs of the associated F_SENDBO are only output again at the RD_BO_xx outputs when a communication error is no longer detected and acknowledgement is made with a positive edge at the ACK_REI input.

Output ACK_REQ = 1 is used to signal that a user acknowledgment at the ACK_REI input is required for the acknowledgment.

 **WARNING**

A user acknowledgement is always required for communication errors

For this, you must interconnect the ACK_REI input with a signal generated by an operator input. An interconnection with an automatically generated signal is not permitted.

Startup characteristics

After start-up of the sending and receiving F-systems, the communication between the connection partners F_SENDBO and F_RCVBO must be established for the first time. The fail-safe values active at the SUBBO_xx inputs are output during this time period. The SUBS_ON output is set to 1.

The SENDMODE output is preset with 0 and is not updated as long as output SUBS_ON = 1.

RETVAL output

| Bit no. | Assignment | Possible error causes | Corrective measures |
|---------|-----------------------------------|--|---|
| Bit 0 | Reserve | — | — |
| Bit 1 | Receiver outputs fail-safe values | See bits 2-7 | Check bits 2-7 |
| Bit 2 | ERROR bit of USEND is set | Basic communication problems of internally called SFB 8 "USEND" are detected | Bit 8-15 = Evaluate "STATUS" of SFB 8 "USEND" or SFB 9 "URCV" |
| | | See also description for bit 7 | See also description for bit 7 |
| Bit 3 | ERROR bit of USEND is set | Basic communication problems of internally called SFB 8 "USEND" are detected | Bit 8-15 = Evaluate "STATUS" of SFB 8 "USEND" or SFB 9 "URCV" |
| | | See also description for bit 7 | See also description for bit 7 |

| Bit no. | Assignment | Possible error causes | Corrective measures |
|-------------|---|--|---|
| Bit 4 | ERROR bit of URCV is set | Basic communication problems of the internally called SFB 9 "USEND" are detected. | Bit 8-15 = Evaluate "STATUS" of SFB 8 "USEND" or SFB 9 "URCV" |
| | | See also description for bit 7 | See also description for bit 7 |
| Bit 5 | CRC error is detected | See description for bit 7 | See description for bit 7 |
| Bit 6 | Sequence number error is detected | See description for bit 7 | See description for bit 7 |
| Bit 7 | Timeout is detected | Connection configuration is incorrect | Check connection configuration and download it again |
| | | Bus connection to the partner F-CPU is faulty | Check the bus connection and ensure that no external sources of disturbance are present. |
| | | F-monitoring time of the F-CPU and of the partner F-CPU is set too short | Check the assigned F-monitoring time TIMEOUT for F_SENDDP and F_RCVDP of both F-CPU's. Set a higher value if necessary. Compile S7 programs again and download them to the F-CPU's. |
| | | STOP or internal error of the CPs | Switch CPs to RUN. Check the diagnostics buffer of the CPs. Replace the CPs, if necessary. |
| | | STOP, partial shutdown or full shutdown or internal error of the F-CPU/partner F-CPU | Switch F-CPU's to RUN. Perform F-startup. Check diagnostics buffer of the F-CPU's. Replace F-CPU's, if necessary. |
| | | The communication was disabled with EN_SEND = 0. | Enable communication again at the associated F_SENDBO with EN_SEND = 1 |
| Bits 8 - 15 | = "STATUS" error information of the internally called SFB 8 "USEND" or SFB 9 "URCV" | See description of the "STATUS" error information in the online help for the SFB 8/SFB 9 or in the " System Software for S7 300/400 System and Standard Functions (http://support.automation.siemens.com/W/view/en/1214574) " | — |
| | | S7 connection has changed, e.g. the IP address of the CP was changed | Compile S7 programs again and download them to the F-CPU's. |

Error handling

An F-STOP is triggered when an error occurs in the safety data format in the associated instance DB. The following diagnostics event is then entered in the diagnostics buffer of the F-CPU:

- "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.2.4 F_SENDR: Sending of 20 data elements of data type F_REAL in a fail-safe manner to another F-CPU


Function

The F_SENDR F-Block sends the data of data type F_REAL pending at the SD_R_xx inputs in a fail-safe manner to another F-CPU. The data must be received there using the F_RCVR F-Block.


To reduce the bus load, you can temporarily shut down communication between the F-CPU's. To do so, supply input EN_SEND with "0" (default = "1"). Send data are then no longer sent to the associated F_RCVR and the assigned fail-safe values are made available to F_SENDR during this time period. If communication was already established between the connection partners, a communication error is detected.

You must specify the local ID of the S7 connection from the perspective of the F-CPU (from the connection table in *NetPro*) at input ID.

Communication between F-CPU's takes place hidden in the background by means of a special safety protocol. You must define a communication association between an F_SENDR in one F-CPU and an F_RCVR in the other F-CPU by assigning an odd number at the R_ID input of the F_SENDR and F_RCVR. Associated F_SENDR and F_RCVR receive the same value for R_ID.

| |
|---|
|  WARNING |
| <p>Value for the relevant address reference</p> <p>The value for each address association (input parameter R_ID; data type: DWORD) is user-defined; however, it must be unique from all other safety-related communication connections in the network. The value R_ID + 1 is internally assigned and must not be used. You must supply inputs ID and R_ID with constant values when calling the F-Block.</p> |

You must assign the desired F-monitoring time at input TIMEOUT. The TIMEOUT input cannot be interconnected.

| |
|--|
|  WARNING |
| <p>Measure and transfer signal level</p> <p>It can be ensured (from a fail-safe standpoint) that a signal level to be transferred will be detected on the sender side and transferred to the receiver only if the signal is pending for at least as long as the assigned F-Monitoring time (TIMEOUT).</p> <p>For information about calculating F-Monitoring times, refer to chapter " Run times, F-Monitoring times, and response times (Page 460) ".</p> |

Note

If the data are received with the F_RCVR F-Block of the *Failsafe Blocks* F-Library (V1_2) or (V1_1), you must assign input EN_SMODE with "0" (default = "1"). Otherwise, a CRC error will be detected by F_RCVR.

In all other cases, you must accept the default setting for input EN_SMODE so that the operating mode of the F-CPU with the F_SENDR can be evaluated at the SENDMODE output of F_RCVR.

Inputs/outputs

| | Name | Data type | Description | Default |
|-----------------|----------|-----------|--|--|
| Inputs: | EN_SEND | BOOL | 1 = ENABLE SEND | 1 |
| | ID | WORD | ADDRESS PARAMETER ID | W#16#0 |
| | R_ID | DWORD | ADDRESS PARAMETER R_ID | DW#16#0 |
| | SD_R_00 | F_REAL | SEND DATA 00 | 0 |
| | ... | | ... | |
| | SD_R_19 | F_REAL | SEND DATA 19 | 0 |
| | CRC_IMP | DWORD | ADDRESS RELATION CRC | DW#16#0 Automatically initialized * |
| | TIMEOUT | F_TIME | F MONITORING TIME | T#0ms |
| | EN_SMODE | F_BOOL | 1 = ENABLE SENDMODE | 1 |
| Outputs: | ERROR | F_BOOL | 1 = COMMUNICATION ERROR | 0 |
| | SUBS_ON | F_BOOL | 1 = SUBSTITUTE VALUES USED FROM RECEIVER | 0 |
| | RETVAL | WORD | ERROR CODE | W#16#0 |

*) Input CRC_IMP is automatically initialized when the S7 program is compiled and must not be changed. When safety programs are compared, input CRC_IMP is indicated as changed if changes have been made to the connection configuration in *NetPro*.

Fail-safe value

Fail-safe values are output from the receiver F_RCVR in the following cases:

- A communication error (e.g., CRC error, timeout) has been detected.
- Communication has been canceled with EN_SEND = 0.
- An F-Startup is pending.

The SUBS_ON output is set to 1.

If the output of the fail-safe value is caused by a communication error, output ERROR = 1 is set additionally.

A "Timeout" communication error is not detected unless communication between the F_SENDR and F_RCVR connection partners has already be established once. If communication cannot be established after startup of the sending and receiving F-Systems,

check the configuration of the safety-related CPU-CPU communication, F_SENDR and F_RCVR parameter assignment, and the bus connection. You can also find possible causes of error by evaluating the RETVAL outputs of the F_SENDR and F_RCVR. In general, you should always evaluate RETVAL of the F_SENDR and F_RCVR because it is possible that only one of the two outputs contains error information.

Reintegration

After a communication error, the data from the receiver pending at the SD_R_xx inputs are only output again if the communication error is no longer detected and the error has been acknowledged with a positive edge at the ACK_REI input of F_RCVR.

Startup characteristics

After the sending and receiving F-Systems are started up, communication must be established initially between the F_SENDR and F_RCVR connection partners. The receiver F_RCVR makes fail-safe values available during this time period. The SUBS_ON output is set to 1.

Output RETVAL

The RETVAL output provides non-fail-safe information on the type of communication errors that occurred for service purposes. You can read out this information on your ES/OS or, if necessary, the information can be evaluated in your standard user program. The DIAG bits are saved until acknowledgment at input ACK_REI of the associated F_RCVR.

Structure of RETVAL

| Bit No. | Assignment | Possible error causes | Remedies |
|---------|--------------------------------------|--|---|
| Bit 0 | Reserve | — | — |
| Bit 1 | SUBSTITUTE VALUES USED FROM RECEIVER | See Bits 2 to 7 | Check Bits 2 to 7 |
| Bit 2 | ERROR bit of USEND set | Basic communication problems detected by internally called SFB 8 "USEND" | Bits 8 to 15 = Evaluate "STATUS" of SFB 8 "USEND" or SFB 9 "URCV" |
| | | See also description for Bit 7 | See also description for Bit 7 |
| Bit 3 | ERROR bit of USEND set | Basic communication problems detected by internally called SFB 8 "USEND" | Bits 8 to 15 = Evaluate "STATUS" of SFB 8 "USEND" or SFB 9 "URCV" |
| | | See also description for Bit 7 | See also description for Bit 7 |
| Bit 4 | ERROR bit of URCV set | Basic communication problems detected by internally called SFB 9 "URCV" | Bits 8 to 15 = Evaluate "STATUS" of SFB 8 "USEND" or SFB 9 "URCV" |
| | | See also description for Bit 7 | See also description for Bit 7 |
| Bit 5 | CRC error detected | See description for Bit 7 | See description for Bit 7 |
| Bit 6 | Sequence number error detected | See description for Bit 7 | See description for Bit 7 |

| Bit No. | Assignment | Possible error causes | Remedies |
|--------------|---|--|--|
| Bit 7 | Timeout detected | Connection configuration is incorrect | Check and reload connection configuration |
| | | Interference in bus connection to partner F-CPU | Check bus connection and ensure that no external interference sources are present |
| | | F-Monitoring time setting for F-CPU and partner F-CPU is too low. | Check assigned F-Monitoring time TIMEOUT at F_SENDR and F_RCVR of both F-CPU. If necessary, set a higher value. Recompile the S7 programs and download them to the F-CPU |
| | | STOP or internal CP fault | Switch CPs to RUN mode Check diagnostic buffer of CPs Replace CPs, if necessary |
| | | STOP, partial or full shutdown, or internal fault in F-CPU or partner F-CPU | Switch F-CPU to RUN mode Perform an F-Startup Check diagnostic buffer of F-CPU Replace F-CPU, if necessary |
| | | Communication was canceled with EN_SEND = 0 | Enable communication again at the associated F_SENDR with EN_SEND = 1 |
| | | S7 connection has changed, the IP address of the CP has changed, for example | Recompile the S7 programs and download them to the F-CPU |
| Bits 8 to 15 | = "STATUS" error information of internally called SFB 8 "USEND" or SFB 9 "URCV" | See description of the "STATUS" error information in the online help for SFB 8/SFB 9 or in Manual " System Software for S7-300/400 System and Standard Functions (http://support.automation.siemens.com/WWW/view/en/1214574) " | — |

Error handling

An F-STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:

- "Safety program: Error in safety data format in DB" (Event ID 16#75DA)


A.2.2.5 F_RCVR: Receiving of 20 data elements of data type F_REAL in a fail-safe manner from another F-CPU

Function


The F-block F_RCVR receives 20 data elements of data type F_REAL from another F-CPU and makes it available to the RD_R_xx outputs. The data must be sent from the other F-CPU with the F-block F_SENDR.

At the ID input you must specify – from the perspective of the F CPU – the local ID of the S7 connection (from connection table in *NetPro*).

Communication between the F-CPU is implemented hidden in the background using a special safety protocol. For this purpose, you must define the communication relationship between an F_SENDR in one F-CPU and an F_RCVR in the other F-CPU by specifying an odd number at the R_ID input of F_SENDR and F_RCVR. Associated F_SENDR and F_RCVR receive the same value for R_ID.

| |
|--|
|  WARNING |
| Value for the respective address relationship |
| The value for the respective address relationship (input parameter R_ID; data type: DWORD) can be selected by the user but must be an odd number and unique network-wide for all safety-related communication connections. The value R_ID + 1 is internally assigned and must not be used. You must supply the ID and R_ID inputs with constant values when the F-block is called. |

You must assign the desired F-monitoring time at the TIMEOUT input. The TIMEOUT input cannot be interconnected.

| |
|---|
|  WARNING |
| Detecting and transmitting the signal level |
| It can only be ensured (from a fail-safe standpoint) that a signal level to be transmitted will be detected on the sender side and transmitted to the receiver if the signal level is present for at least as long as the assigned F-monitoring time (TIMEOUT). |
| For information regarding the calculation of the F-monitoring time, see section " Run times, F-Monitoring times, and response times (Page 460) ". |

The operating mode of the F-CPU with F_SENDR is provided at the SENDMODE output. If the F-CPU with F_SENDR is in deactivated safety mode, the SENDMODE output becomes = 1.

Note

If the data is received from an F_SENDR block from older F-libraries, you must assign the COMMVER_USED input with 0. Otherwise, sequence number errors may occur. Default setting is "1".

I/Os

| | Name | Data type | Explanation | Default |
|---------|---------|-----------|---------------------------|--------------------------------------|
| Inputs: | ID | WORD | Addressing parameter ID | W#16#0 |
| | R_ID | DWORD | Addressing parameter R_ID | W#16#0 |
| | CRC_IMP | DWORD | Address relationship CRC | W#16#0 Is supplied automatically* |

| | Name | Data type | Explanation | Default |
|-----------------|--------------|-----------|---|---------|
| | TIMEOUT | F_TIME | F-monitoring time in ms | T#0ms |
| | ACK_REI | F_BOOL | Acknowledgment for reintegration | 0 |
| | SUBR_00 | F_REAL | Fail-safe value for receive data element 00 | 0 |
| | ... | | ... | |
| | SUBR_19 | F_REAL | Fail-safe value for receive data element 19 | 0 |
| | COMMVER_USED | INT | 0 = Communication with blocks from older F-libraries < V1.3 SP2 (compatibility mode) 1 = Communication with blocks of F-Library V1.3 SP2 (or higher) | 1 |
| Outputs: | | | | |
| | ACK_REQ | BOOL | Acknowledgment for reintegration is required | 0 |
| | ERROR | F_BOOL | 1 = Communication error | 0 |
| | SUBS_ON | F_BOOL | 1 = Fail-safe values are output | 0 |
| | RD_R_00 | F_REAL | Receive data element 00 | 0 |
| | ... | | ... | |
| | RD_R_19 | F_REAL | Receive data element 19 | 0 |
| | SENDMODE | F_BOOL | 1 = F-CPU with F_SENDR in deactivated safety mode | |
| | RETVAl | WORD | Error code | W#16#0 |

*) The CRC_IMP input is automatically supplied when the S7 program is compiled and must not be changed. The CRC_IMP input is indicated as changed during comparison of safety programs if changes were made to the connection configuration in *NetPro*.

Fail-safe values

The fail-safe values active at the SUBR_xx inputs are output in the following cases:

- A communication error (e.g. CRC error, Timeout) was detected.
- The communication was disabled at the associated F_SENDR via EN_SEND = 0.
- An F-startup is present.

The SUBS_ON output is set to 1.

While output SUBS_ON = 1, the SENDMODE output is not updated.

If the output of the fail-safe value is caused by a communication error, output ERROR = 1 is additionally set.


A "Timeout" communication error is detected for the first time when the communication between the F_SENDR and F_RCVR connection partners has already been established once. If communication cannot be established after startup of the sending and receiving F-system, check the configuration of the safety-related CPU-CPU communication, the parameter assignment of F_SENDR and F_RCVR and the bus connection. You can also obtain information on possible error causes by evaluating the RETVAL outputs of F_SENDR

and F_RCVR. In general, always evaluate RETVAL of F_SENDR and F_RCVR as it may be that only one of the two outputs contains error information.

Reintegration

After a communication error, the data active at the SD_R_xx inputs of the associated F_SENDR are only output again at the RD_R_xx outputs when a communication error is no longer detected and acknowledgement is made with a positive edge at the ACK_REI input.

Output ACK_REQ = 1 is used to signal that a user acknowledgment at the ACK_REI input is required for the acknowledgment.

| |
|---|
|  WARNING |
| A user acknowledgement is always required for communication errors. For this, you must interconnect the ACK_REI input with a signal generated by an operator input. An interconnection with an automatically generated signal is not permitted. |

Startup characteristics

After start-up of the sending and receiving F-systems, the communication between the connection partners F_SENDR and F_RCVR must be established for the first time. The fail-safe values active at the SUBR_xx inputs are output during this time period. The SUBS_ON output is set to 1.

The SENDMODE output is preset with 0 and is not updated as long as output SUBS_ON = 1.

RETVAl output

Non fail-safe information about the nature of the communication error that occurred is made available at the RETVAL output for service purposes. You can read out this information on your ES/OS or evaluate it in your standard user program if necessary. The DIAG bits remain stored until you acknowledge at the ACK_REI input.

Structure of RETVAL

| Bit no. | Assignment | Possible error causes | Corrective measures |
|---------|-----------------------------------|--|---|
| Bit 0 | Reserve | — | — |
| Bit 1 | Receiver outputs fail-safe values | See bits 2-7 | See bits 2-7 |
| Bit 2 | ERROR bit of USEND is set | Basic communication problems of internally called SFB 8 "USEND" are detected | Bit 8-15 = Evaluate "STATUS" of SFB 8 "USEND" or SFB 9 "URCV" |
| | | See also description for bit 7 | See also description for bit 7 |
| Bit 3 | ERROR bit of USEND is set | Basic communication problems of internally called SFB 8 "USEND" are detected | Bit 8-15 = Evaluate "STATUS" of SFB 8 "USEND" or SFB 9 "URCV" |
| | | See also description for bit 7 | See also description for bit 7 |

| Bit no. | Assignment | Possible error causes | Corrective measures |
|-------------|---|--|---|
| Bit 4 | ERROR bit of URCV is set | Basic communication problems of the internally called SFB 9 "USEND" are detected. | Bit 8-15 = Evaluate "STATUS" of SFB 8 "USEND" or SFB 9 "URCV" |
| | | See also description for bit 7 | See also description for bit 7 |
| Bit 5 | CRC error is detected | See description for bit 7 | See description for bit 7 |
| Bit 6 | Sequence number error is detected | See description for bit 7 | See description for bit 7 |
| Bit 7 | Timeout is detected | Connection configuration is incorrect | Check connection configuration and download it again |
| | | Bus connection to the partner F-CPU is faulty | Check the bus connection and ensure that no external sources of disturbance are present. |
| | | F-monitoring time of the F-CPU and of the partner F-CPU is set too short | Check the assigned F-monitoring time TIMEOUT for F_SENDR and F_RCVR of both F-CPU's. Set a higher value if necessary. Compile S7 programs again and download them to the F-CPU's. |
| | | STOP or internal error of the CPs | Switch CPs to RUN. Check the diagnostics buffer of the CPs. Replace the CPs, if necessary. |
| | | STOP, partial shutdown or full shutdown or internal error of the F-CPU/partner F-CPU | Switch F-CPU's to RUN. Perform F-startup. Check diagnostics buffer of the F-CPU's. Replace F-CPU's, if necessary. |
| | | The communication was disabled with EN_SEND = 0. | Enable communication again at the associated F_SENDR with EN_SEND = 1 |
| | | S7 connection has changed, e.g. the IP address of the CP was changed | Compile S7 programs again and download them to the F-CPU's. |
| Bits 8 - 15 | = "STATUS" error information of the internally called SFB 8 "USEND" or SFB 9 "URCV" | See description of the "STATUS" error information in the online help for the SFB 8/SFB 9 or in the " System Software for S7 300/400 System and Standard Functions (http://support.automation.siemens.com/W/view/en/1214574) " | — |

Error handling

An F-STOP is triggered when an error occurs in the safety data format in the associated instance DB. The following diagnostics event is then entered in the diagnostics buffer of the F-CPU:

- "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.2.6 F_SDS_BO: Sending of 32 data elements of data type F_BOOL in a fail-safe manner to another F-CPU

Function

The F_SDS_BO F-Block sends the data of data type F_BOOL pending at the SD_BO_xx inputs in a fail-safe manner to another F-CPU. The data must be received there using the F_RDS_BO F-Block.


Note

The F_SDS_BO F-Block can also send the data of data type F_BOOL pending at the SD_BO_xx inputs in a fail-safe manner to another F-CPU with *S7 Distributed Safety*. The data must be received there with the F_RCVS7 F-Block and an F-Communication DB with exactly 32 data elements of data type F_BOOL.


To reduce the bus load, you can temporarily shut down communication between the F-CPU's. To do so, supply input EN_SEND with "0" (default = "1"). Send data are then no longer sent to the associated F_RDS_BO and the assigned fail-safe values are made available to F_RDS_BO during this time period. If communication was already established between the connection partners, a communication error is detected.

You must specify the local ID of the S7 connection from the perspective of the F-CPU (from the connection table in *NetPro*) at input ID.

Communication between F-CPU's takes place hidden in the background by means of a special safety protocol. You must define a communication association between an F_SDS_BO in one F-CPU and an F_RDS_BO in the other F-CPU by assigning an odd number at the R_ID input of the F_SDS_BO and F_RDS_BO. Associated F_SDS_BO and F_RDS_BO receive the same value for R_ID.

| |
|---|
|  WARNING |
| Value for the relevant address reference |
| The value for each address association (input parameter R_ID; data type: DWORD) is user-defined; however, it must be unique from all other safety-related communication connections in the network. The value R_ID + 1 is internally assigned and must not be used. You must supply inputs ID and R_ID with constant values when calling the F-Block. |

You must assign the desired F-monitoring time at input TIMEOUT. The TIMEOUT input cannot be interconnected.

| |
|---|
|  WARNING |
| Measure and transfer signal level |
| It can be ensured (from a fail-safe standpoint) that a signal level to be transferred will be detected on the sender side and transferred to the receiver only if the signal is pending for at least as long as the assigned F-Monitoring time (TIMEOUT). |
| For information about calculating F-Monitoring times, refer to chapter " Run times, F-Monitoring times, and response times (Page 460) ". |

Inputs/outputs

| | Name | Data type | Description | Default |
|-----------------|----------|-----------|--|---|
| Inputs: | EN_SEND | BOOL | 1 = ENABLE SEND | 1 |
| | ID | WORD | ADDRESS PARAMETER ID | W#16#0 |
| | R_ID | DWORD | ADDRESS PARAMETER R_ID | DW#16#0 |
| | SD_BO_00 | F_BOOL | SEND DATA 00 | 0 |
| | ... | | ... | |
| | SD_BO_31 | F_BOOL | SEND DATA 31 | 0 |
| | CRC_IMP | DWORD | ADDRESS RELATION CRC | DW#16#0 Automatically initial- ized * |
| | TIMEOUT | F_TIME | F MONITORING TIME in ms | T#0ms |
| Outputs: | ERROR | F_BOOL | 1 = COMMUNICATION ERROR | 0 |
| | SUBS_ON | F_BOOL | 1 = SUBSTITUTE VALUES USED FROM RECEIVER | 0 |
| | RETVAl | WORD | ERROR CODE | W#16#0 |

*) Input CRC_IMP is automatically initialized when the S7 program is compiled and must not be changed. When safety programs are compared, input CRC_IMP is indicated as changed if changes have been made to the connection configuration in *NetPro*.

Fail-safe values

Fail-safe values are output from the receiver F_RDS_BO in the following cases:

- A communication error (e.g., CRC error, timeout) has been detected.
- Communication has been canceled with EN_SEND = 0.
- An F-Startup is pending.

The SUBS_ON output is set to 1.

If the output of the fail-safe value is caused by a communication error, output ERROR = 1 is set additionally.

A "Timeout" communication error is not detected unless communication between the F_SDS_BO and F_RDS_BO connection partners has already be established once. If communication cannot be established after startup of the sending and receiving F-Systems, check the configuration of the safety-related CPU-CPU communication, F_SDS_BO and F_RDS_BO parameter assignment, and the bus connection. You can also find possible causes of error by evaluating the RETVAL outputs of the F_SDS_BO and F_RDS_BO. In general, you should always evaluate RETVAL of the F_SDS_BO and F_RDS_BO because it is possible that only one of the two outputs contains error information.

Reintegration

After a communication error, the data from the receiver pending at the SD_BO_xx inputs are only output again if the communication error is no longer detected and the error has been acknowledged with a positive edge at the ACK_REI input of F_RDS_BO.

Startup characteristics

After the sending and receiving F-Systems are started up, communication must be established initially between the F_SDS_BO and F_RDS_BO connection partners. The receiver F_RDS_BO makes fail-safe values available during this time period. The SUBS_ON output is set to 1.

Output RETVAL

The RETVAL output provides non-fail-safe information on the type of communication errors that occurred for service purposes. You can read out this information on your ES/OS or, if necessary, the information can be evaluated in your standard user program. The DIAG bits are saved until acknowledgment at input ACK_REI of the associated F_RDS_BO.

Structure of RETVAL

| Bit No. | Assignment | Possible error causes | Remedies |
|---------|--------------------------------------|--|---|
| Bit 0 | Reserve | — | — |
| Bit 1 | SUBSTITUTE VALUES USED FROM RECEIVER | See Bits 2 to 7 | Check Bits 2 to 7 |
| Bit 2 | ERROR bit of USEND set | Basic communication problems detected by internally called SFB 8 "USEND" | Bits 8 to 15 = Evaluate "STATUS" of SFB 8 "USEND" or SFB 9 "URCV" |
| | | See also description for Bit 7 | See also description for Bit 7 |
| Bit 3 | ERROR bit of USEND set | Basic communication problems detected by internally called SFB 8 "USEND" | Bits 8 to 15 = Evaluate "STATUS" of SFB 8 "USEND" or SFB 9 "URCV" |
| | | See also description for Bit 7 | See also description for Bit 7 |
| Bit 4 | ERROR bit of URCV set | Basic communication problems detected by internally called SFB 9 "URCV" | Bits 8 to 15 = Evaluate "STATUS" of SFB 8 "USEND" or SFB 9 "URCV" |
| | | See also description for Bit 7 | See also description for Bit 7 |
| Bit 5 | CRC error detected | See description for Bit 7 | See description for Bit 7 |
| Bit 6 | Sequence number error detected | See description for Bit 7 | See description for Bit 7 |

| Bit No. | Assignment | Possible error causes | Remedies |
|--------------|---|--|--|
| Bit 7 | Timeout detected | Connection configuration is incorrect | Check and reload connection configuration |
| | | Interference in bus connection to partner F-CPU | Check bus connection and ensure that no external interference sources are present |
| | | F-monitoring time setting for F-CPU and partner F-CPU is too low. | Check assigned F-monitoring time TIMEOUT at F_SDS_BO and F_RDS_BO of both F-CPU's. If necessary, set a higher value. Recompile the S7 programs and load them to the F-CPU's. |
| | | STOP or internal CP fault | Switch CPs to RUN mode. Check diagnostic buffer of CPs. Replace CPs, if necessary. |
| | | STOP, partial or full shutdown, or internal fault in F-CPU or partner F-CPU | Switch F-CPU's to RUN mode. Perform an F-Startup. Check diagnostic buffer of F-CPU's. Replace F-CPU's, if necessary. |
| | | Communication was canceled with EN_SEND = 0. | Enable communication again at the associated F_SDS_BO with EN_SEND = 1. |
| | | S7 connection has changed, the IP address of the CP has changed, for example. | Recompile the S7 programs and download them to the F-CPU's. |
| Bits 8 to 15 | = "STATUS" error information of internally called SFB 8 "USEND" or SFB 9 "URCV" | See description of the "STATUS" error information in the online help for SFB 8/SFB 9 or in Manual " System Software for S7-300/400 System and Standard Functions (http://support.automation.siemens.com/WW/view/en/1214574) " | — |

Error handling

An F-STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:

- "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.2.7 F_RDS_BO: Receiving of 32 data elements of data type F_BOOL in a fail-safe manner from another F-CPU

Function

The F-block F_RDS_BO receives 32 data elements of data type F_BOOL from another F-CPU and makes it available to the RD_BO_xx outputs. The data must be sent from the other F-CPU with the F-block F_SDS_BO.

Note

The F-block F_RDS_BO can also receive the 32 data elements of data type F_BOOL in a fail-safe manner from an F-CPU with *S7 Distributed Safety*. The data must then be sent there with the F-block F_SENDS7 and an F-communication DB with exactly 32 data elements of data type F_BOOL.

At the ID input you must specify – from the perspective of the F CPU – the local ID of the S7 connection (from connection table in *NetPro*).

Communication between the F-CPU's is implemented hidden in the background using a special safety protocol. For this purpose, you must define the communication relationship between an F_SDS_BO in one F-CPU and an F_RDS_BO in the other F-CPU by specifying an odd number at the R_ID input of F_SDS_BO and F_RDS_BO. Associated F_SDS_BO and F_RDS_BO receive the same value for R_ID.

| |
|--|
|  WARNING |
|--|

| |
|--|
| Value for the respective address relationship |
|--|

| |
|--|
| The value for the respective address relationship (input parameter R_ID; data type: DWORD) can be selected by the user but must be an odd number and unique network-wide for all safety-related communication connections. The value R_ID + 1 is internally assigned and must not be used. You must supply the ID and R_ID inputs with constant values when the F-block is called. |
|--|

You must assign the desired F-monitoring time at the TIMEOUT input. The TIMEOUT input cannot be interconnected.

| |
|--|
|  WARNING |
|--|

| |
|--|
| Detecting and transmitting the signal level |
|--|

| |
|---|
| It can only be ensured (from a fail-safe standpoint) that a signal level to be transmitted will be detected on the sender side and transmitted to the receiver if the signal level is present for at least as long as the assigned F-monitoring time (TIMEOUT). |
|---|

| |
|---|
| For information regarding the calculation of the F-monitoring time, see section " Run times, F-Monitoring times, and response times (Page 460) ". |
|---|

The operating mode of the F-CPU with F_SDS_BO is provided at the SENDMODE output. If the F-CPU with F_SDS_BO is in deactivated safety mode, the SENDMODE output becomes = 1.

Note

If the data is received from an F_SDS_BO block from older F-libraries, you must assign the COMMVER_USED input with 0. Otherwise, sequence number errors may occur. Default setting is "1".

I/Os

| | Name | Data type | Explanation | Default |
|-----------------|--------------|-----------|---|--|
| Inputs: | ID | WORD | Addressing parameter ID | W#16#0 |
| | R_ID | DWORD | Addressing parameter R_ID | DW#16#0 |
| | CRC_IMP | DWORD | Address relationship CRC | DW#16#0 To be automatically supplied* |
| | TIMEOUT | F_TIME | F-monitoring time in ms | T#0ms |
| | ACK_REI | F_BOOL | Acknowledgement for reintegration | 0 |
| | SUBBO_00 | F_BOOL | Fail-safe value for receive data element 00 | 0 |
| | ... | | ... | |
| | SUBBO_31 | F_BOOL | Fail-safe value for receive data element 31 | 0 |
| | COMMVER_USED | INT | 0 = Communication with blocks from older F-libraries < V1.3 SP2 (compatibility mode) 1 = Communication with blocks of F-Library V1.3 SP2 (or higher) | 1 |
| Outputs: | ACK_REQ | BOOL | Acknowledgement for reintegration is required | 0 |
| | ERROR | F_BOOL | 1 = Communication error | 0 |
| | SUBS_ON | F_BOOL | 1 = Fail-safe values are output | 0 |
| | RD_BO_00 | F_BOOL | Receive data element 00 | 0 |
| | ... | | ... | |
| | RD_BO_31 | F_BOOL | Receive data element 31 | 0 |
| | SENDMODE | F_BOOL | 1 = F-CPU with F_SDS_BO in deactivated safety mode | 0 |
| | RETVAL | WORD | Error code | W#16#0 |

*) The CRC_IMP input is automatically supplied when the S7 program is compiled and must not be changed. The CRC_IMP input is indicated as changed during comparison of safety programs if changes were made to the connection configuration in *NetPro*.

Fail-safe values

The fail-safe values active at the SUBBO_XX inputs are output in the following cases:

- A communication error (e.g. CRC error, Timeout) was detected.
- The communication was disabled at the associated F_SDS_BO via EN_SEND = 0.
- An F-startup is present.

The SUBS_ON output is set to 1.

While output SUBS_ON = 1, the SENDMODE output is not updated.


If the output of the fail-safe value is caused by a communication error, output ERROR = 1 is additionally set.

A "Timeout" communication error is detected for the first time when the communication between the F_SDS_BO and F_RDS_BO connection partners has already been established once. If communication cannot be established after startup of the sending and receiving F-systems, check the configuration of the safety-related CPU-CPU communication, the parameter assignment of F_SDS_BO and F_RDS_BO and the bus connection. You can also obtain information on possible error causes by evaluating the RETVAL outputs of F_SDS_BO and F_RDS_BO. In general, always evaluate RETVAL of F_SDS_BO and F_RDS_BO as it may be that only one of the two outputs contains error information.

Reintegration

After a communication error, the data active at the SD_BO_XX inputs of the associated F_SDS_BO are only output again at the RD_BO_XX outputs when a communication error is no longer detected and acknowledgement is made with a positive edge at the ACK_REI input.

Output ACK_REQ = 1 is used to signal that a user acknowledgment at the ACK_REI input is required for the acknowledgment.

| |
|---|
|  WARNING |
| A user acknowledgement is always required for communication errors |
| For this, you must interconnect the ACK_REI input with a signal generated by an operator input. An interconnection with an automatically generated signal is not permitted. |

Startup characteristics

After start-up of the sending and receiving F-systems, the communication between the connection partners F_SDS_BO and F_RDS_BO must be established for the first time. The fail-safe values active at the SUBBO_XX inputs are output during this time period. The SUBS_ON output is set to 1.

The SENDMODE output is preset with 0 and is not updated as long as output SUBS_ON = 1.

RETVAl output

Non fail-safe information about the nature of the communication error that occurred is made available at the RETVAL output for service purposes. You can read out this information on your ES/OS or evaluate it in your standard user program if necessary. The DIAG bits remain stored until you acknowledge at the ACK_REI input.

Structure of RETVAL

| Bit no. | Assignment | Possible error causes | Corrective measures |
|---------|-----------------------------------|--|--|
| Bit 0 | Reserve | — | — |
| Bit 1 | Receiver outputs fail-safe values | See bits 2-7 | Check bits 2-7 |
| Bit 2 | ERROR bit of USEND is set | Basic communication problems of internally called SFB 8 "USEND" are detected | Bit 8-15 = Evaluate "STATUS" of SFB 8 "USEND" or SFB 9 "URCV" |
| | | See also description for bit 7 | See also description for bit 7 |
| Bit 3 | ERROR bit of USEND is set | Basic communication problems of internally called SFB 8 "USEND" are detected. | Bit 8-15 = Evaluate "STATUS" of SFB 8 "USEND" or SFB 9 "URCV" |
| | | See also description for bit 7 | See also description for bit 7 |
| Bit 4 | ERROR bit of URCV is set | Basic communication problems of the internally called SFB 9 "USEND" are detected. | Bit 8-15 = Evaluate "STATUS" of SFB 8 "USEND" or SFB 9 "URCV" |
| | | See also description for bit 7 | See also description for bit 7 |
| Bit 5 | CRC error is detected | See description for bit 7 | See description for bit 7 |
| Bit 6 | Sequence number error is detected | See description for bit 7 | See description for bit 7 |
| Bit 7 | Timeout is detected | Connection configuration is incorrect | Check connection configuration and download it again |
| | | Bus connection to the partner F-CPU is faulty | Check the bus connection and ensure that no external sources of disturbance are present. |
| | | F-monitoring time of the F-CPU and of the partner F-CPU is set too short | Check the assigned F-monitoring time TIMEOUT for F_SDS_BO and F_RDS_BO of both F-CPUs. Set a higher value if necessary. Compile S7 programs again and download them to the F-CPUs. |
| | | STOP or internal error of the CPs | Switch CPs to RUN. Check the diagnostics buffer of the CPs. Replace the CPs, if necessary. |
| | | STOP, partial shutdown or full shutdown or internal error of the F-CPU/partner F-CPU | Switch F-CPUs to RUN. Perform F-startup. Check diagnostics buffer of the F-CPUs. Replace F-CPUs, if necessary. |

| Bit no. | Assignment | Possible error causes | Corrective measures |
|-------------|---|---|--|
| | | The communication was disabled with EN_SEND = 0. | Enable communication again at the associated F_SDS_BO with EN_SEND = 1 |
| | | S7 connection has changed, e.g. the IP address of the CP was changed | Compile S7 programs again and download them to the F-CPU's. |
| Bits 8 - 15 | = "STATUS" error information of the internally called SFB 8 "USEND" or SFB 9 "URCV" | See description of the "STATUS" error information in the online help for the SFB 8/SFB 9 or in the " System Software for S7 300/400 System and Standard Functions (http://support.automation.siemens.com/WW/view/en/1214574) " | — |

Error handling

An F_STOP is triggered when an error occurs in the safety data format in the associated instance DB. The following diagnostics event is then entered in the diagnostic buffer of the F-CPU:

- "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.3 F-Blocks for comparing two input values of the same type

Overview

| Block name | Block number | Description |
|------------|--------------|---|
| F_CMP_R | FB 313 | Comparator for two REAL values |
| F_LIM_HL | FB 314 | Monitoring of upper limit violation of a REAL value |
| F_LIM_LL | FB 315 | Monitoring of lower limit violation of a REAL value |

A.2.3.1 F_CMP_R Comparator for two REAL values

Function

This F-Block compares two inputs of data type F_REAL and sets outputs GT, GE, EQ, LT or LE to "1", whatever the comparator result:

- GT = 1 if IN1 > IN2
- GE = 1 if IN1 ≥ IN2
- EQ = 1 if IN1 = IN2
- LT = 1 if IN1 < IN2
- LE = 1 if IN1 ≤ IN2

Inputs/outputs

| | Name | Data type | Description | Default |
|----------|------|-----------|-------------|---------|
| Inputs: | IN1 | F_REAL | Input 1 | 0 |
| | IN2 | F_REAL | Input 2 | 0 |
| Outputs: | GT | F_BOOL | IN1 > IN2 | 0 |
| | GE | F_BOOL | IN1 ≥ IN2 | 0 |
| | EQ | F_BOOL | IN1 = IN2 | 0 |
| | LT | F_BOOL | IN1 < IN2 | 0 |
| | LE | F_BOOL | IN1 ≤ IN2 | 0 |

Error handling

- If one of the inputs IN1 or IN2 is an invalid floating point number (NaN), outputs GT and LT are set to 1.
- An F-STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.3.2 F_LIM_HL: Monitoring of upper limit violation of a REAL value

Function

This F-Block monitors the input variable U for limit violation (U_HL). A hysteresis can also be specified at the HYS input to avoid fluttering of the QH output in the event of fluctuations in the input value.

- $U \geq U_{HL}$: If the upper limit is exceeded, output QH = 1.
- $(U_{HL} - HYS) \leq U < U_{HL}$: QH remains unchanged in this range.
- $U < (U_{HL} - HYS)$: If the limit value hysteresis is fallen below, output QH = 0.

The QHN output corresponds to the negated QH output.

The limit value and hysteresis are also available as non-fail-safe data at the U_HL_O and HYS_O outputs for further processing in the standard user program.

Inputs/outputs

| | Name | Data type | Description | Default |
|---------|---------|-----------|------------------|---------|
| Inputs: | U | F_REAL | INPUT | 0.0 |
| | U_HL | F_REAL | UPPER LIMIT | 100.0 |
| | HYS | F_REAL | HYSTERESIS | 0.0 |
| | SUBS_IN | F_BOOL | SUBSTITUTE VALUE | 0 |

| | Name | Data type | Description | Default |
|-----------------|--------|-----------|---------------------------|---------|
| Outputs: | QH | F_BOOL | 1 = UPPER LIMIT VIOLATION | 0 |
| | QHN | F_BOOL | NEGATING OUTPUT QH | 1 |
| | U_HL_O | REAL | UPPER LIMIT | 100.0 |
| | HYS_O | REAL | HYSTERESIS | 0.0 |

Error handling

- If one of the inputs U, U_HL or HYS is an invalid floating point number (NaN) or if invalid floating-point numbers (NaN) arise due to calculations in the F-Block, the fail-safe value at the input SUBS_IN is output at output QH.

If invalid floating-point numbers (NaNs) arise due to calculations in the F-Block, the following diagnostic event is then entered in the diagnostic buffer of the F-CPU:

- "Safety program: invalid REAL number in DB" (Event ID 16#75D9)

- An F_STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:

- "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.3.3 F_LIM_LL: Monitoring of lower limit violation of a REAL value

Function

This F-Block monitors the input variable U for lower limit violation

(U_LL). A hysteresis can also be specified at the HYS input to avoid fluttering of the QL output in the event of fluctuations in the input value.

- $U \leq U_LL$: If the lower limit is violated, output QL = 1.
- $U_LL < U \leq (U_LL + HYS)$: QL remains unchanged in this range.
- $U > (U_LL + HYS)$: If the upper limit is exceeded violated + hysteresis, output QL = 0.

Output QLN corresponds to the negated QL output.

The limit value and hysteresis are also available as non-fail-safe data at the U_LL_O and HYS_O outputs for evaluation in the standard user program.

Inputs/outputs

| | Name | Data type | Description | Default |
|----------------|---------|-----------|-----------------|---------|
| Inputs: | U | F_REAL | INPUT | 0.0 |
| | U_LL | F_REAL | LOWER LIMIT | 100.0 |
| | HYS | F_REAL | HYSTERESIS | 0.0 |
| | SUBS_IN | F_BOOL | FAIL-SAFE VALUE | 0 |

| | Name | Data type | Description | Default |
|-----------------|--------|-----------|---------------------------|---------|
| Outputs: | QL | F_BOOL | 1 = LOWER LIMIT VIOLATION | 0 |
| | QLN | F_BOOL | NEGATING OUTPUT QL | 1 |
| | U_LL_O | REAL | LOWER LIMIT | 100.0 |
| | HYS_O | REAL | HYSTERESIS | 0.0 |

Error handling

- If one of the inputs U, U_LL or HYS is an invalid floating point number (NaN) or if invalid floating-point numbers (NaNs) arise due to calculations in the F-Block, the fail-safe value at the input SUBS_IN is output at output QL.

If invalid floating-point numbers (NaNs) arise due to calculations in the F-Block, the following diagnostic event is then entered in the diagnostic buffer of the F-CPU:

- "Safety program: invalid REAL number in DB" (Event ID 16#75D9)
- An F_STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.4 Voter blocks for inputs of data type REAL and BOOL

Overview

| Block name | Block number | Description |
|------------|--------------|---|
| F_2oo3DI | FB 316 | 2oo3 evaluation of inputs of data type BOOL with discrepancy analysis |
| F_2oo3AI | FB 318 | 2oo3 evaluation of inputs of data type REAL with discrepancy analysis |
| F_1oo2AI | FB 317 | 1oo2 evaluation of inputs of data type REAL with discrepancy analysis |

A.2.4.1 F_2oo3DI: 2oo3 evaluation of inputs of data type BOOL with discrepancy analysis

Function

This F-block monitors three binary inputs for signal state 1. The OUT output is 1 when at least two INx inputs are 1. Otherwise, the OUT output is 0. The OUTN output corresponds to the negated OUT output.

If input DIS_ON = 1 is set, a discrepancy analysis is performed. If one INx input differs from the other two INy inputs longer than the assigned discrepancy time DIS_TIME, a discrepancy error is detected and saved with 1 in the DIS and DIS_D outputs.

If a discrepancy is no longer detected, the discrepancy error is acknowledged according to the parameter assignment of ACK_NEC:


- When ACK_NEC = 0, an automatic acknowledgment is carried out.
- When ACK_NEC = 1, you must acknowledge the discrepancy error with a positive edge at the ACK input.

Output ACK_REQ = 1 is used to signal that user acknowledgment at the ACK input is required for acknowledging the discrepancy error.

I/Os

| | Name | Data type | Explanation | Default |
|-----------------|----------|-----------|-----------------------------|---------|
| Inputs: | IN1 | F_BOOL | Input 1 | 0 |
| | IN2 | F_BOOL | Input 2 | 0 |
| | IN3 | F_BOOL | Input 3 | 0 |
| | DIS_ON | F_BOOL | 1 = Discrepancy analysis | 0 |
| | DIS_TIME | F_TIME | Discrepancy time in ms | 1000 |
| | ACK_NEC | F_BOOL | 1 = Acknowledgment required | 0 |
| | ACK | F_BOOL | Acknowledgment | 0 |
| Outputs: | OUT | F_BOOL | Output | 0 |
| | OUTN | F_BOOL | Output inverted | 1 |
| | DIS | F_BOOL | Discrepancy error | 0 |
| | DIS_D | BOOL | DATA component of DIS | 0 |
| | ACK_REQ | BOOL | Acknowledgement required | 0 |

Fail-safe user times

| |
|--|
|  WARNING |
| <p>When determining your response times when using an F-block with time monitoring, take into account the following timing uncertainties:</p> <ul style="list-style-type: none"> • The timing uncertainty familiar from the standard program that arises due to the cyclic processing • The tolerance of the internal monitoring of the times in the F-CPU <ul style="list-style-type: none"> – For time values from 10 ms to 50 s: 5 ms – For time values from > n × 50 s to (n+1) × 50 s: ± (n+1) × 5 ms |

Error handling

An F-STOP is triggered when there is an error in the safety data format in the associated instance DB. The following diagnostics event is then entered in the diagnostics buffer of the F-CPU:

- "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.4.2 F_2oo3AI: 2oo3 evaluation of inputs of the REAL data type with discrepancy analysis

Function

This F-block performs a 2oo3 evaluation of REAL values with discrepancy analysis.

This block is generally intended for detecting the failure or discrepancy of a sensor.

If a REAL value is invalid, a 1oo2 evaluation is performed. It calculates the average and median or the maximum and minimum of the INx inputs, depending on the QBADx inputs:

- If all INx inputs are valid (QBAD1, QBAD2 and QBAD3 = 0) and no discrepancy error was saved (DIS1CH = 0, DISALL = 0), the average $[(IN1+IN2+IN3)/3]$ is provided at the OUT_AVG output and the median of IN1, IN2 and IN3 is provided at the MED_MAX and MED_MIN outputs.
- If all INx inputs are valid (QBAD1, QBAD2 and QBAD3 = 0) and the assigned tolerance DELTA is exceeded at all INx inputs (DIS1CH = 0, DISALL = 1), the average $[(IN1+IN2+IN3)/3]$ is provided at the OUT_AVG output and the median of IN1, IN2 and IN3 is provided at the MED_MAX and MED_MIN outputs.
- If all INx inputs are valid (QBAD1, QBAD2 and QBAD3 = 0), the assigned tolerance DELTA is exceeded at all INx inputs and a discrepancy error was saved (DIS1CH = 1, DISALL = 1), then the outputs behave as follows:
 - MODE = 0
 - OUT_AVG = Average of the INx inputs that were previously discrepancy-free, i.e. when DIS1CH = 1 and DISALL = 0.
 - MED_MAX = MED_MIN = Median of IN1, IN2 and IN3
 - MODE = 1
 - OUT_AVG = Average of the INx inputs that were previously discrepancy-free, i.e. when DIS1CH = 1 and DISALL = 0.
 - MED_MAX = MED_MIN = Median of IN1, IN2 and IN3
 - MODE = 3
 - OUT_AVG = Average of the INx inputs that were previously discrepancy-free, i.e. when DIS1CH = 1 and DISALL = 0.
 - MED_MAX = Maximum of the INx inputs that were previously discrepancy-free
 - MED_MIN = Minimum of the INx inputs that were previously discrepancy-free

- If all INx inputs are valid (QBAD1, QBAD2 and QBAD3 = 0) and a discrepancy error was saved (DIS1CH = 1, DISALL = 0), then the outputs behave as follows:
 - MODE = 0
OUT_AVG = Average of the INx inputs that were previously discrepancy-free
MED_MAX = MED_MIN = Median of IN1, IN2 and IN3
 - MODE = 1
OUT_AVG = Average of the discrepancy-free INx inputs
MED_MAX = MED_MIN = Median of IN1, IN2 and IN3
 - MODE = 3
OUT_AVG = Average of the discrepancy-free INx inputs
MED_MAX = Maximum of the discrepancy-free INx inputs
MED_MIN = Minimum of the discrepancy-free INx inputs
-

Note

Change of the MODE parameter

Changes to the MODE parameter are only possible by carrying out a cold restart of the CPU. An online change is not permitted.

A cold restart of the CPU is not allowed under PCS 7. For this reason, the MODE parameter can be changed in PCS 7 by means of a full download with changed parameters.

- If only two inputs INx are valid (QBADx = 0 und QBADy = 1), the average of the valid INx inputs is provided at the OUT_AVG output, the maximum and minimum of the valid INx inputs are provided at the MED_MAX and MED_MIN outputs, respectively, and QBAD_1CH = 1 is set.
- If only one INx input is valid (QBADx = 0 and QBADy = 1), INx is provided at the OUT_AVG, MED_MAX and MED_MIN outputs and QBAD_2CH = 1 is set.
- If no INx input is valid (QBAD1, QBAD2 and QBAD3 = 1), the fail-safe value SUBS_V is provided at the OUT_AVG, MED_MAX and MED_MIN outputs and QBAD_ALL = 1 is set.

A discrepancy analysis is carried out as follows:

- All INx inputs are valid (QBAD1, QBAD2 and QBAD3 = 0)
 - If one INx input differs from the two other INy inputs by more than the assigned tolerance DELTA and for longer than the assigned discrepancy time DIS_TIME, a discrepancy error is detected and saved with 1 in the DIS1CH and DIS1CH_D outputs.
 - If all INx inputs differ by more than the assigned tolerance DELTA and for longer than the assigned discrepancy time DIS_TIME, a discrepancy error is detected and saved with 1 in the DISALL and DISALL_D outputs.
- Two inputs INx are valid (QBADx = 0 and QBADy = 1):
 - If the two valid INx inputs differ by more than the assigned tolerance DELTA and for longer than the assigned discrepancy time DIS_TIME, a discrepancy error is detected and saved with 1 in the DISALL and DISALL_D outputs.
- Only one INX input is valid (QBADx = 0 and QBADy = 1) or all INx inputs are invalid (QBAD1, QBAD2 and QBAD3 = 1):
 - A discrepancy analysis is not carried out.

The absolute value is always used in this case for the DELTA und DIS_TIME inputs.

Note

If the process signals at the inputs fluctuate strongly, you must set the DELTA und DIS_TIME parameters in such a way that regular fluctuations between the process values are not detected as errors.

If the inputs fall within the assigned tolerance again, the discrepancy error is acknowledged depending on the parameter assignment of ACK_NEC:

- When ACK_NEC = 0, an automatic acknowledgment is carried out.
- When ACK_NEC = 1, you must acknowledge the discrepancy error with a positive edge at the ACK input.

Output ACK_REQ = 1 is used to signal that user acknowledgment at the ACK input is required for acknowledging the discrepancy error.

Note

If you want to implement a trigger of your safety function when a limit is exceeded (e.g. with F-block F_LIM_HL), you must use the MED_MAX output for the limit monitoring. If you want to implement a trigger of your safety function when a limit is fallen below (e.g. with F-block F_LIM_LL), you must use the MED_MIN output for the limit monitoring.

You may only then use the OUT_AVG output if it flows into an evaluation in which – dependent on the process situation – the safe direction is represented once by the maximum and once by the minimum. In this case, output DISALL = 1 should also trigger the safety function.

I/Os

| | Name | Data type | Explanation | Default |
|-----------------|----------|-----------|---|---------|
| Inputs: | DELTA | F_REAL | Tolerance between INx | 0.0 |
| | DIS_TIME | F_TIME | Discrepancy time in ms | 1000 |
| | IN1 | F_REAL | Input 1 | 0.0 |
| | IN2 | F_REAL | Input 2 | 0.0 |
| | IN3 | F_REAL | Input 3 | 0.0 |
| | QBAD1 | F_BOOL | 1 = IN1 input invalid | 0 |
| | QBAD2 | F_BOOL | 1 = IN2 input invalid | 0 |
| | QBAD3 | F_BOOL | 1 = IN3 input invalid | 0 |
| | SUBS_V | F_REAL | Fail-safe value | 0.0 |
| | ACK_NEC | F_BOOL | 1 = Acknowledgment required | 0 |
| | ACK | F_BOOL | Acknowledgment | 0 |
| | MODE | F_WORD | <p>Operating mode</p> <ul style="list-style-type: none"> • MODE = 0: Discrepancies that occur are saved and taken into account when calculating OUT_AVG. Detected discrepancy errors are saved until all three process values are within the assigned tolerance DELTA again. Only then does OUT_AVG correspond to the average of all 3 process values again. (Behavior in S7 F Systems Lib <= V1_3 SP1) • MODE = 1: Discrepancy errors that occurred previously are not saved. Only current discrepancies are taken into account when calculating OUT_AVG. (New behavior starting in S7 F Systems Lib V1_3 SP2). • MODE = 2: Reserved, not supported • MODE = 3: Discrepancy errors that occurred previously are not saved. Only current discrepancies are taken into account when calculating OUT_AVG. MED_MAX corresponds to the maximum of the discrepancy-free INx inputs. MED_MIN corresponds to the minimum of the discrepancy-free INx inputs. (New behavior starting in S7 F Systems Lib V1_3 SP2) | 0 |
| Outputs: | OUT_AVG | F_REAL | Average of INx | 0.0 |
| | MED_MAX | F_REAL | Median/maximum of INx | 0.0 |


| | Name | Data type | Explanation | Default |
|--|----------|-----------|-------------------------------------|---------|
| | MED_MIN | F_REAL | Median/minimum of INx | 0.0 |
| | QBAD_1CH | F_BOOL | One INx input invalid | 0 |
| | QBAD_2CH | F_BOOL | Two INx inputs invalid | 0 |
| | QBAD_ALL | F_BOOL | All INx inputs invalid | 0 |
| | DIS1CH | F_BOOL | Discrepancy error at one INx input | 0 |
| | DISALL | F_BOOL | Discrepancy error at all INx inputs | 0 |
| | DIS1CH_D | BOOL | DATA component of DIS1CH | 0 |
| | DISALL_D | BOOL | DATA component of DISALL | 0 |
| | ACK_REQ | BOOL | Acknowledgement required | 0 |
| | Q_MODE | WORD | Status of MODE input | 0 |

Use together with F-channel driver F_CH_AI

If you interconnect the INx input of F_2oo3AI with the V output of F_CH_AI, you must observe the following:

- Interconnect the QBADx input of F_2oo3AI with the QBAD output of the F_CH_AI whose V output you want to interconnect with the INx input of F_2oo3AI.

Fail-safe user times

| |
|--|
|  WARNING |
| <p>When determining your response times when using an F-block with time monitoring, take into account the following timing uncertainties:</p> <ul style="list-style-type: none"> • The timing uncertainty familiar from the standard program that arises due to the cyclic processing • The tolerance of the internal monitoring of the times in the F-CPU <ul style="list-style-type: none"> – For time values from 10 ms to 50 s: 5 ms – For time values from $> n \times 50$ s to $(n+1) \times 50$ s: $\pm (n+1) \times 5$ ms |

Error handling

- If an INx input is an invalid floating-point number (NaN), it is handled as an invalid INx input with QBAD = 1.
- If the DELTA input is an invalid floating-point number (NaN), DIS1CH, DISALL, DIS1CH_D and DISALL_D are set to 1.

- If calculations result in invalid floating-point numbers (NaN) in the F-block, the fail-safe value SUBS_V is provided at the OUT_AVG, MED_MAX and MED_MIN outputs, QBAD_1CH, QBAD_2CH and QBAD_ALL = 1 is set, and the following diagnostics event is entered in the diagnostics buffer of the F-CPU:
 - "Safety program: Invalid REAL number in DB" (Event ID 16#75D9).
- An F-STOP is triggered when there is an error in the safety data format in the associated instance DB. The following diagnostics event is then entered in the diagnostics buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.4.3 F_1oo2AI: 1oo2 evaluation of inputs of data type REAL with discrepancy analysis

Function

This F-Block carries out a 1oo2 evaluation of REAL values using discrepancy analysis. It calculates the mean value, the maximum and the minimum of inputs IN1 and IN2 depending on inputs QBADx:

- If both INx inputs are valid (QBAD1 and QBAD2 = 0), the IN1 and IN2 mean $[(IN1+IN2)/2]$ is made available at output OUT_AVG, the maximum at output OUT_MAX and the minimum at output OUT_MIN.
- If only input INx is valid (QBADx = 0 and QBADy = 1), INx is made available at outputs OUT_AVG, OUT_MAX and OUT_MIN and QBAD_1CH = 1 is set.
- If no input INx is valid (QBAD1 and QBAD2 = 1), the SUBS_V the fail-safe value is made available at outputs OUT_AVG, OUT_MAX and OUT_MIN and QBAD_ALL = 1 is set.

If both inputs INx are valid (QBAD1 and QBAD2 = 0), a discrepancy analysis is carried out:

If the INx inputs discrepancy is greater than the assigned DELTA tolerance and longer than the assigned discrepancy time DIS_TIME, a discrepancy error is detected and stored at the outputs DIS and DIS_D with 1. The absolute value is always used for the DELTA and DIS_TIME inputs.

When the assigned tolerance is adhered to again, the discrepancy error is acknowledged according to the ACK_NEC parameter assignment:

- If ACK_NEC = 0, the acknowledgment is automatic.
- If ACK_NEC = 1 you must acknowledge the discrepancy error with a rising edge at input ACK.

The ACK_REQ = 1 output signals that a user acknowledgment is necessary at input ACK to acknowledge the discrepancy error.

Note

If you want to implement safety function triggering when an upper limit is exceeded (e.g. with F-Block F_LIM_HL), you must use output OUT_MAX for limit violation monitoring. If you want to implement safety function triggering when a lower limit is violated (e.g. with F-Block F_LIM_LL), you must use output OUT_MIN for limit violation monitoring.

Output OUT_AVG may only be used in evaluations in which - depending on the process situation - the maximum and the minimum each represent the safe direction once. In this case, output DIS = 1 should also trigger the safety function.

Inputs/outputs


| | Name | Data type | Description | Default |
|-----------------|----------|-----------|------------------------------|---------|
| Inputs: | DELTA | F_REAL | Tolerance between INx | 0.0 |
| | DIS_TIME | F_TIME | Discrepancy time in ms | 0 |
| | IN1 | F_REAL | Input 1 | 0.0 |
| | IN2 | F_REAL | Input 2 | 0.0 |
| | QBAD1 | F_BOOL | 1 = Input IN1 invalid | 0 |
| | QBAD2 | F_BOOL | 1 = Input IN2 invalid | 0 |
| | SUBS_V | F_REAL | SUBSTITUTE VALUE | 0.0 |
| | ACK_NEC | F_BOOL | 1 = ACKNOWLEDGMENT NECESSARY | 0 |
| | ACK | F_BOOL | ACKNOWLEDGMENT | 0 |
| Outputs: | OUT_AVG | F_REAL | AVERAGE VALUE OF INx | 0.0 |
| | OUT_MAX | F_REAL | MAXIMUM VALUE OF INx | 0.0 |
| | OUT_MIN | F_REAL | MINIMUM VALUE OF INx | 0.0 |
| | QBAD_1CH | F_BOOL | ONE INPUT INx INVALID | 0 |
| | QBAD_ALL | F_BOOL | ALL INPUTS INx INVALID | 0 |
| | DIS | F_BOOL | DISCREPANCY ERROR | 0 |
| | DIS_D | BOOL | DISCREPANCY ERROR DATA | 0 |
| | ACK_REQ | BOOL | ACKNOWLEDGMENT REQUEST | 0 |

Used together with F-Channel driver F_CH_AI

If you interconnect input INx of the F_1oo2AI with output V of an F_CH_AI, you must observe the following:

- Interconnect input QBADx of the F_1oo2AI with the QBAD output of the F_CH_AI and its output V with input INx of the F_1oo2AI.

Fail-safe user times

| |
|---|
|  WARNING |
| <p>When using an F-Block with time processing, take the following timing imprecision sources into account when determining your response times:</p> <ul style="list-style-type: none"> • Known timing imprecision (based on standard systems) resulting from cyclic processing • Tolerance of internal time monitoring in the F-CPU <ul style="list-style-type: none"> – For time values between 10 ms and 50 s: 5 ms – For time values from $> n \times 50 \text{ s}$ to $(n+1) \times 50 \text{ s}$: $\pm (n+1) \times 5 \text{ ms}$ |

Error handling

- If an input INx is an invalid floating point number (NaN), it is treated as an invalid input INx with QBADx = 1.
- If the DELTA input is an invalid floating point number (NaN), DIS and DIS_D are set to 1.
- If invalid floating-point numbers (NaNs) arise due to calculations in the F-Block, the fail-safe value SUBS_V is made available at the outputs OUT_AVG, OUT_MAX and OUT_MIN, QBAD_1CH and QBAD_ALL = 1 are set and the following diagnostic event is entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Invalid REAL number in DB" (Event ID 16#75D9).
- An F-STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA).

A.2.5 Blocks and F-Blocks for data conversion

A.2.5.1 Blocks and F-blocks for data conversion

Overview

F-blocks


| Block name | Block number | Description |
|------------|--------------|---|
| F_SWC_CB | FB 614 | Processing of a parameter of data format F_BOOL for operator input via the OS (Change process values) |
| F_SWC_CR | FB 615 | Processing of a parameter of data format F_REAL for operator input via the OS (Change process values) |
| F_SWC_P | FB 335 | Centralized control of operator input via the OS (Change process values, Maintenance Override, Fail-safe acknowledgement) |

| Block name | Block number | Description |
|------------|--------------|--|
| F_SWC_BO | FB 336 | Processing of a parameter of data type F_BOOL for operator input via the OS (Maintenance Override, Fail-safe acknowledgment) |
| F_SWC_R | FB 337 | Processing of a parameter of data type F_REAL for operator input via the OS (Maintenance Override) |
| F_FR_FDI | FB 339 | Conversion from F_REAL to F_DINT |
| F_FDI_FR | FB 340 | Conversion from F_DINT to F_REAL |
| F_BO_FBO | FB 361 | Conversion from BOOL to F_BOOL |
| F_R_FR | FB 362 | Conversion from REAL to F_REAL |
| F_QUITES | FB 367 | Fail-safe acknowledgement via the ES/OS |
| F_TI_FTI | FB 368 | Conversion from TIME to F_TIME |
| F_I_FI | FB 369 | Conversion from INT to F_INT |
| F_FI_FR | FB 460 | Conversion from F_INT to F_REAL |
| F_FR_FI | FB 461 | Conversion from F_REAL to F_INT |
| F_CHG_R | FB 478 | Safety Data Write for F_REAL |
| F_CHG_BO | FB 479 | Safety Data Write for F_BOOL |

Blocks

| Block name | Block number | Description |
|------------|--------------|--|
| F_FBO_BO | FC 303 | Conversion from F_BOOL to BOOL |
| F_FR_R | FC 304 | Conversion from F_REAL to REAL |
| F_FI_I | FC 305 | Conversion from F_INT to INT |
| F_FTI_TI | FC 306 | Conversion from F_TIME to TIME |
| SWC_CHG | FB 482 | Operator function for "Change process values" |
| SWC_MOS | FB 338 | Operator function for "Maintenance Override" |
| SWC_QOS | FB 481 | Operator function for "Fail-safe acknowledgment" |

Validity check

|  WARNING |
|--|
| <p>Validity check</p> <p>The F-blocks F_BO_FBO, F_I_FI, F_TI_FTI and F_R_FR only convert data. For this reason, you must program additional measures for validity checks in the safety program.</p> |

The simplest type of validity check is a range specification with fixed high limit and low limit, e.g. with F_LIM_R.

Not all input parameters can be checked for validity in a sufficiently easy manner.


A.2.5.2 F_SWC_CB: Processing of a parameter of data format F-BOOL for operator input via the OS

Function


The F-block F_SWC_CB enables changes to be made to F-parameters of data type F_BOOL in the safety program of the F-CPU from an OS (Change process values).

The OUT output is interconnected in the safety program with the I/O whose value is to be changed.

If a change was made on the faceplate in the required operator sequence within the time assigned for MAX_TIME of the F_SWC_P, the value entered on the faceplate is made available at the OUT output.

| |
|--|
|  WARNING |
| The "Secure Write Command++" functionality allows changes to the safety program to be made during RUN mode |
| As a result, the following additional safety measures are required: |
| <ul style="list-style-type: none">• Ensure that changes that may compromise plant safety cannot be made. You can use the EN_SWC input of the F-block F_SWC_P for this purpose, for example, by controlling it with a keyswitch or on a process-dependent basis via the safety program.• Ensure that only authorized persons can make changes. |
| Examples: |
| <ul style="list-style-type: none">• Control the EN_SWC input of the F-block F_SWC_P with a keyswitch.• Set up access protection for the operator stations where the "Secure Write Command++" function can be performed. |

When input CS_MODE = 1, the value made available at the OUT output is applied to the CS_VAL input, and output CHANGED = 1 is set.

| |
|---|
|  WARNING |
| The CHANGED output cannot be evaluated in the safety program |
| CHANGED = 1 merely indicates that a change at the OUT output has been transferred to the CS_VAL input. |
| If the value changed using the "Secure Write Command++" function is to be in effect after a cold restart, you must manually correct the value at the CS_VAL input in the offline program and load memory. |

I/Os

| | Name | Data type | Explanation | Default |
|-----------------|---------|-----------|--|---------|
| Inputs: | CS_VAL | F_BOOL | Initial value for OUT at a cold restart | 0 |
| | CS_MODE | F_BOOL | Switch for CS_VAL 1 = Apply changed OUT to CS_VAL 0 = CS_VAL remains unchanged | 0 |
| | WS_MODE | F_BOOL | Switch for warm restart 1 = At a warm restart/RESTART for F_SHUTDOWN, the value at the OUT output is retained 0 = At a warm restart/RESTART for F_SHUTDOWN, OUT receives the value of CS_VAL | 1 |
| Outputs: | | | | |
| | OUT | F_BOOL | Current value of the operator-controlled parameter | 0 |
| | AKT_VAL | BOOL | Current value of the operator-controlled parameter for the OS | 0 |
| | CHANGED | BOOL | Indication of whether CS_VAL was changed 1 = CS_VAL was changed via the HMI | 0 |
| | CS_USED | F_BOOL | Indicates the value that was made available for OUT after startup. 1 = CS_VAL 0 = Last valid value | 0 |

Note

The interconnection of the AKT_VAL output establishes the connection to the OS.

 **WARNING**

Interconnection of the CS_VAL input is not permitted.

Startup behavior


After an F-startup, the F-block behaves as follows:

- After a CPU STOP followed by a cold restart of the F-CPU or during initial run:
In the first cycle after a cold restart or during the initial run, the value assigned at the CS_VAL input is made available at the OUT output. The CS_USED output is set to 1. CS_USED is reset to 0 as soon as "Secure Write Command++" has been successfully completed for the first time.
- After a CPU STOP followed by a warm restart of the F-CPU or after an F-STOP followed by a positive edge at the RESTART input of the F-block F_SHUTDOWN:
In the first cycle after a warm restart or after a positive edge at the RESTART input of the F-block F_SHUTDOWN, the last valid value of OUT is made available at the OUT output

when input WS_MODE = 1. The CS_USED output retains its default value (0). When input WS_MODE = 0, the F-block behaves the same as after a cold restart.

Note

Prior to the initial processing of the F-block after an F-startup, the default value is at the OUT and CS_USED outputs.

| |
|---|
|  WARNING |
| F-startup After an F-startup, plant safety must not be compromised due to either the presence of the CS_VAL value at the OUT output or the presence of the last valid value at the OUT output. If necessary, evaluate the CS_USED output to determine whether the CS_VAL value or the last valid value was made available at the OUT output after an F-startup. You are not permitted to change the default value "0" of CS_USED. If a warm restart is performed after a cold restart, CS_USED is reset to the default value "0", even if the CS_VAL value is currently present at the OUT output. |

Error handling

An F_STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostics event is then entered in the diagnostics buffer of the F-CPU:

- "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

See also

SWC_CHG: Operator function for Change process values (Page 306)

A.2.5.3 F_SWC_CR: Processing of a parameter of data format F-REAL for operator input via the OS

Function

The F-block F_SWC_CR enables changes to be made to F-parameters of data type F_REAL in the safety program of the F-CPU from an OS (Change process values).


The OUT output is interconnected in the safety program with the I/O whose value is to be changed.

The limits for the change are specified using the MIN and MAX inputs.

The maximum increment of the change is specified at the MAXDELTA input.


If a change was made on the faceplate in the required operator sequence within the time assigned for MAX_TIME of the F_SWC_P, the value entered on the faceplate is made available at the OUT output, provided it meets the following conditions:

- The value is within the limits assigned for the MIN and MAX inputs.
- The maximum increment of the change assigned for the MAXDELTA input is not exceeded.

| |
|---|
|  WARNING |
| <p>The "Secure Write Command++" functionality allows changes to the safety program to be made during RUN mode.</p> <p>As a result, the following additional safety measures are required:</p> <ul style="list-style-type: none"> • Ensure that changes that may compromise plant safety cannot be made. You can use the EN_SWC input of the F-block F_SWC_P for this purpose, for example, by controlling it with a keyswitch or on a process-dependent basis via the safety program. • Ensure that only authorized persons can make changes. <p>Examples:</p> <ul style="list-style-type: none"> • Control the EN_SWC input of the F-block F_SWC_P with a keyswitch. • Set up access protection for the operator stations where the "Secure Write Command++" function can be performed. |

As an alternative to the measures above, select the MIN and MAX inputs in such a way that values that could compromise plant safety cannot be specified using the "Change process data" function.

When input CS_MODE = 1, the value made available at the OUT output is applied to the CS_VAL input, and output CHANGED = 1 is set.

| |
|--|
|  WARNING |
| <p>The CHANGED output cannot be evaluated in the safety program.</p> <p>CHANGED = 1 merely indicates that a change at the OUT output has been transferred to the CS_VAL input.</p> <p>If the value changed using the "Secure Write Command++" function is to be in effect after a cold restart, you must manually correct the value at the CS_VAL input in the offline program and load memory.</p> |


I/Os

| | Name | Data type | Explanation | Default |
|----------------|----------|-----------|--|---------|
| Inputs: | MIN | F_REAL | Low limit for F-parameter change | 0.0 |
| | MAX | F_REAL | High limit for F-parameter change | 100.0 |
| | MAXDELTA | F_REAL | Maximum change between the current value (OUT) and the new value | 10.0 |

| | Name | Data type | Explanation | Default |
|-----------------|---------|-----------|--|---------|
| | CS_VAL | F_REAL | Initial value for OUT at a cold restart | 0 |
| | CS_MODE | F_BOOL | Switch for CS_VAL 1 = Apply changed OUT to CS_VAL 0 = CS_VAL remains unchanged | 0 |
| | WS_MODE | F_BOOL | Switch for warm restart 1 = At a warm restart/RESTART for F_SHUTDOWN, the value at the OUT output is retained 0 = At a warm restart/RESTART for F_SHUTDOWN, OUT receives the value of CS_VAL | 1 |
| Outputs: | | | | |
| | OUT | F_BOOL | Current value of the operator-controlled parameter | 0 |
| | AKT_VAL | F_BOOL | Current value of the operator-controlled parameter for the OS | 0 |
| | CHANGED | BOOL | Indication of whether CS_VAL was changed 1 = CS_VAL was changed via the HMI. | 0 |
| | CS_USED | F_BOOL | Indicates the value that was made available for OUT after startup. 1 = CS_VAL 0 = Last valid value | 0 |
| | CURR_R | REAL | Copy of OUT.DATA Here, the unit of measurement to be displayed in the faceplate can be assigned using the "Unit" I/O property. | 0.0 |

Note

The interconnection of the AKT_VAL output establishes the connection to the OS.

| |
|--|
|  WARNING |
| The CS_VAL, MIN, MAX and MAXDELTA inputs must not be interconnected. |

Startup behavior

After an F-startup, the F-block behaves as follows:

- After a CPU STOP followed by a cold restart of the F-CPU or during initial run:

In the first cycle after a cold restart or during the initial run, the value assigned at the CS_VAL input is made available at the OUT output. The CS_USED output is set to 1. CS_USED is reset to 0 as soon as "Secure Write Command++" has been successfully completed for the first time.
- After a CPU STOP followed by a warm restart of the F-CPU or after an F-STOP followed by a positive edge at the RESTART input of the F-block F_SHUTDOWN:

In the first cycle after a warm restart or after a positive edge at the RESTART input of the F-block F_SHUTDOWN, the last valid value of OUT is made available at the OUT output when input WS_MODE = 1. The CS_USED output retains its default value ("0"). When input WS_MODE = 0, the F-block behaves the same as after a cold restart.
- During startup, OUT and ACT_VAL are initialized with the cold restart value CS_VAL if this value is within the MIN and MAX limits. If CS_VAL < MIN, OUT and AKT_VAL are initialized with the MIN value. If CS_VAL > MAX, OUT and AKT_VAL are initialized with the MAX value.

Note

Prior to the initial processing of the F-block after an F-startup, the default value is at the OUT and CS_USED outputs.

| |
|--|
|  WARNING |
|--|

F-startup

After an F-startup, plant safety must not be compromised due to either the presence of the CS_VAL value at the OUT output or the presence of the last valid value at the OUT output.

If necessary, evaluate the CS_USED output to determine whether the CS_VAL value or the last valid value was made available at the OUT output after an F-startup. You are not permitted to change the default value "0" of CS_USED. If a warm restart is performed after a cold restart, CS_USED is reset to the default value "0", even if the CS_VAL value is currently present at the OUT output.

Principle

An F_STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostics event is then entered in the diagnostics buffer of the F-CPU:

- "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

See also

SWC_CHG: Operator function for Change process values (Page 306)

A.2.5.4 F_SWC_P: Centralized control of operator input via the OS

Function

This F-block performs the protocol handling with the OS for controlling the F_BOOL and F_REAL parameters. For this purpose, it implements a special safety protocol and monitors the required operator sequence. It has no dependency on the function behind the operator input. At least one F_SWC_P must be placed for each F-shutdown group so that one or more operator functions (SWC_CHG, SWC_MOS, SWC_QOS) can be controlled.

For the "Change process values", "Maintenance Override", and "Fail-safe acknowledgment" functions, you must assign an identifier for the utilized F-CPU that is unique from all others in the system. There are two ways of doing this:


- Assign the IDENT input for the F-block F_SWC_P
- Assign the identifier to the HID of the F-CPU

The identifier at the IDENT input has precedence. If you assign the identifier to the HID of the F-CPU and you do not use the IDENT input, the IDENT input remains empty when the program is compiled.

Use of a keyswitch

To ensure that only authorized persons perform operator inputs via the OS, you can connect the EN_SWC input of the F-block F_SWC_P to a keyswitch.

Input EN_SWC = 'true' must be set during an operator input. If EN_SWC = 'false' is reset after an operator input, all existing bypasses will be deactivated. However, set fail-safe values are retained.

| |
|---|
|  WARNING |
| <p>The "Secure Write Command++" functionality allows changes to the safety program to be made during RUN mode</p> <p>As a result, the following additional safety measures are required:</p> <ul style="list-style-type: none">• Identification of the F-CPU must be unique system-wide. S7 F Systems uses the IDENT parameter of F_SWC_P or the HID of the F-CPU.• Ensure that changes that may compromise plant safety cannot be made. You can use the EN_SWC input of the F-block F_SWC_P for this purpose, for example, by controlling it with a keyswitch or on a process-dependent basis via the safety program.• Ensure that only authorized persons can make changes. <p>Examples:</p> <ul style="list-style-type: none">– Control the EN_SWC input of the F-block F_SWC_P with a keyswitch.– Set up access protection for the operator stations where the "Secure Write Command++" function can be executed. |

I/Os

| | Name | Data type | Explanation | Default |
|-----------------|----------|-------------|---|---------|
| Inputs: | EN_SWC | F_BOOL | Key-operated switch: 0=No operator input allowed 1=Operator input allowed | 0 |
| | IDENT | STRING [32] | CPU identifier | " |
| | MAX_TIME | F_TIME | Maximum duration of an operator input, Timeout time | 1 m |
| Outputs: | ADR_OSPA | DWORD | Connection between protocol and operator control block | 0 |

A.2.5.5 F_SWC_BO: Processing of a parameter of data type F_BOOL for operator input via the OS

Function

The F-block F_SWC_BO enables changes to be made to F-parameters of data type F_BOOL in the safety program of the F-CPU from an OS using "Maintenance Override" or "Fail-safe acknowledgment".

The OUT output is interconnected in the safety program with the I/O whose value is to be changed.

If a change was made on the faceplate in the required operator sequence within the time assigned for MAX_TIME of the F_SWC_P, the value entered on the faceplate is made available at the OUT output.


- Interconnection with SWC_MOS ("Maintenance Override"):

The S and R inputs can be used to set or reset the OUT and AKT_VAL outputs independent of an operator input. OUT and AKT_VAL are set using the positive edge at S. Resetting has priority, so the resetting occurs as long as R = 1. The setting of OUT and AKT_VAL is also possible when the keyswitch is not active because this is relevant only for a bypass by the operator input (soft bypass).

S and R can be used as a hard bypass for connection of a sensor. This always has precedence over the soft bypass controlled via the OS. For this reason, an active operator input is cancelled when the hard bypass is active.

- Interconnection with SWC_QOS ("Fail-safe acknowledgment"):

Operator control of the S and R inputs is only possible via the OS. Setting and resetting by the running program is not supported.


| |
|--|
|  WARNING |
| <p>The "Secure Write Command++" functionality allows changes to the safety program to be made during RUN mode.</p> <p>As a result, the following additional safety measures are required:</p> <ul style="list-style-type: none"> • Ensure that changes that may compromise plant safety cannot be made. You can use the EN_SWC input of the F-block F_SWC_P for this purpose, for example, by controlling it with a keyswitch or on a process-dependent basis via the safety program. • Ensure that only authorized persons can make changes. Examples: <ul style="list-style-type: none"> – Control the EN_SWC input of the F-block F_SWC_P with a keyswitch. – Set up access protection for the operator stations where the "Secure Write Command++" function can be performed. |

I/Os

| | Name | Data type | Explanation | Default |
|-----------------|---------|-----------|---|---------|
| Inputs: | S | F_BOOL | Set input | 0 |
| | R | F_BOOL | Reset input | 0 |
| | CS_VAL | F_BOOL | Cold restart | 0 |
| Outputs: | OUT | F_BOOL | Current value of the operator-controlled parameter | 0 |
| | AKT_VAL | BOOL | Current value of the operator-controlled parameter for the OS | 0 |


Note

The interconnection of the AKT_VAL output establishes the connection to the OS.

| |
|--|
|  WARNING |
| Interconnection of the CS_VAL input is not permitted. |

Startup behavior

During startup, OUT and ACT_VAL are initialized with the value of CS_VAL at a cold restart.

| |
|---|
|  WARNING |
| <p>F-startup</p> <p>After an F-startup, plant safety must not be compromised due to the presence of the CS_VAL value at the OUT and AKT_VAL outputs.</p> |

Error handling

- An F_STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostics event is then entered in the diagnostics buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

See also

SWC_MOS: Command function for Maintenance Override (Page 307)

A.2.5.6 F_SWC_R: Processing of a parameter of data type F_REAL for operator input via the OS

Function


The F-block F_SWC_CR enables changes to be made to F-parameters of data type F_REAL in the safety program of the F-CPU from an OS using "Maintenance Override".

The OUT output is interconnected in the safety program with the I/O whose value is to be changed.

The limits for the change are specified using the MIN and MAX inputs.

If a change was made on the faceplate in the required operator sequence within the time assigned for MAX_TIME of the F_SWC_P, the value entered on the faceplate is made available at the OUT output, provided it meets the following conditions:


- The value is within the limits assigned for the MIN and MAX inputs.

| |
|---|
|  WARNING |
| <p>The "Secure Write Command++" functionality allows changes to the safety program to be made during RUN mode.</p> <p>As a result, the following additional safety measures are required:</p> <ul style="list-style-type: none"> • Ensure that changes that may compromise plant safety cannot be made. You can use the EN_SWC input of the F-block F_SWC_P for this purpose, for example, by controlling it with a keyswitch or on a process-dependent basis via the safety program. • Ensure that only authorized persons can make changes. <p>Examples:</p> <ul style="list-style-type: none"> – Control the EN_SWC input of the F-block F_SWC_P with a keyswitch. – Set up access protection for the operator stations where the "Secure Write Command++" function can be performed. |

As an alternative to the measures above, select the MIN and MAX inputs in such a way that values that could compromise plant safety cannot be specified using the "Maintenance Override" function.

I/Os

| | Name | Data type | Explanation | Default |
|-----------------|---------|-----------|---|---------|
| Inputs: | MIN | F_REAL | Minimum value for IN | 0.0 |
| | MAX | F_REAL | Maximum value for IN | 100.0 |
| | CS_VAL | F_REAL | Cold restart value | 0.0 |
| Outputs: | OUT | F_REAL | Current value of the operator-controlled parameter | 0.0 |
| | AKT_VAL | REAL | Current value of the operator-controlled parameter for the OS | 0.0 |


| |
|--|
|  WARNING |
| The CS_VAL, MIN and MAX inputs must not be interconnected. |

Note

The interconnection of the AKT_VAL output establishes the connection to the OS.

Startup behavior

During startup, OUT and ACT_VAL are initialized with the cold restart value CS_VAL if this value is within the MIN and MAX limits. If CS_VAL < MIN, OUT and AKT_VAL are initialized with the MIN value. If CS_VAL > MAX, OUT and AKT_VAL are initialized with the MAX value.

| |
|--|
|  WARNING |
| F-startup |
| After an F-startup, plant safety must not be compromised due to the presence of the CS_VAL value at the OUT and AKT_VAL outputs. |

Error handling

- An F_STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostics event is then entered in the diagnostics buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

See also

SWC_MOS: Command function for Maintenance Override (Page 307)

A.2.5.7 F_FR_FDI: Conversion from F_REAL to F_DINT**Function**

This F-Block converts the F_REAL F-Data type at the IN input to the F_DINT F-Data type at the OUT output.

Following the conversion of F_REAL to F_DINT, if the value at the IN input exceeds the upper limit that can be portrayed by the F_INT data type, 2,147,483,647 is output at the OUT output and output OUTU is set to 1. At F_DINT values greater than (>) 2,147,483,583, the range is already exceeded.

If the range is undershot (IN is less than (<) the F_DINT value that can be portrayed), the smallest F_DINT value of -2,147,483,648 is output at output OUT, and output OUTL is set to 1.

Inaccuracies/rounding

If the value at input IN is located outside the range -16777216,0 to 16777215,0, it is possible for the output value to be rounded in F_DINT format, as values in the F_REAL format require 8 bits of the 32-bit real value to represent the exponent.

Inputs/outputs

| | Name | Data type | Description | Default |
|-----------------|------|-----------|------------------------------|---------|
| Input: | IN | F_REAL | Input | 0.0 |
| Outputs: | OUT | F_DINT | Output | 0 |
| | OUTU | F_BOOL | Upper number range violation | 0 |
| | OUTL | F_BOOL | Lower number range violation | 0 |

Error handling

- If the IN input is an invalid floating point number (NaN), 0 is output at the OUT output and OUTU and OUTL are set to 1.
- An F-STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.5.8 F_FDI_FR: Conversion from F_DINT to F_REAL

Function

This F-Block converts the F_DINT F-Data type at the IN input to the F_REAL F-Data type at the OUT output.

Inaccuracies/rounding

If the value at input IN is greater than (>) 16,777,215 or less than (<) -16,777,216, this can result in an inaccuracy in the output value of 127, maximum, compared to the input value. That is, the value in F_DINT format is rounded up or rounded off for representation in F_REAL format, as 8 bits of the 32-bit real value are required to represent the exponent. If the value is rounded off, RND_OFF = 1 is set. If the value is rounded up, RND_UP = 1 is set.

If values at input IN are greater than or equal to (>=) 2,147,483,584, the output value of data type F_REAL is always rounded up. In this case, RND_UP = 1 is always set.

Inputs/outputs

| | Name | Data type | Description | Default |
|-----------------|---------|-----------|-------------------------------------|---------|
| Input: | IN | F_DINT | Input | 0 |
| Outputs: | OUT | F_REAL | Output | 0.0 |
| | RND_UP | F_BOOL | Output value is a rounded-up value | 0 |
| | RND_OFF | F_BOOL | Output value is a rounded-off value | 0 |

Error handling

An F-STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:

- "Safety program: Error in safety data format in DB" (Event ID 16#75DA).

A.2.5.9 F_BO_FBO: Conversion from BOOL to F_BOOL

Function

This F-Block converts the BOOL data type at the IN input to the corresponding F_BOOL F-Data type at the OUT output.

This enables signals formed in the standard user program to be evaluated in the safety program following a validity check.

Inputs/outputs

| | Name | Data type | Description | Default |
|---------|------|-----------|-------------|---------|
| Input: | IN | BOOL | Input | 0 |
| Output: | OUT | F_BOOL | Output | 0 |

Error handling

None

A.2.5.10 F_R_FR: Conversion from REAL to F_REAL**Function**

This F-Block converts the REAL data type at the IN input to the corresponding F_REAL F-Data type at the OUT output.

This enables signals formed in the standard user program to be evaluated in the safety program following a validity check (using F-Block F_LIM_R, for example).

Inputs/outputs

| | Name | Data type | Description | Default |
|---------|------|-----------|-------------|---------|
| Input: | IN | REAL | Input | 0.0 |
| Output: | OUT | F_REAL | Output | 0.0 |

Error handling

None

A.2.5.11 F_QUITES: Fail-safe acknowledgement via the ES/OS**Function**


This F-Block enables fail-safe acknowledgment from a non-fail-safe ES/OS. This allows reintegration of F-I/O to be controlled via the ES/OS, for example. Acknowledgment takes place in two steps:

1. Changing the IN input to the value "6"
2. Changing the IN input from the value "6" to the value "9" within one minute

The F-Block evaluates whether or not the value at input IN changes to "9" within **1 second at the earliest** or **1 minute at the latest** after the value changes to "6". The signal "1" is then output at the OUT output (output for acknowledgement) for the duration of one cycle.

If an invalid value is entered or if the change in the value to "9" occurs before 1 second or after 1 minute has elapsed, the IN input is reset to 0 and the two steps indicated above have to be repeated.

During the time in which the change from "6" to "9" must occur, the non-fail-safe Q output is set to 1. Otherwise, Q has a value of 0.

| |
|--|
|  WARNING |
| Reintegration through User Acknowledgment with F_QUITES |
| The two acknowledgment steps must not be triggered by a single operation, for example, by automatically storing them along with the time conditions in one program requiring a single operation to trigger them. By programming separate acknowledgment steps, you prevent erroneous triggering of an acknowledgment by your non-fail-safe operator station. |

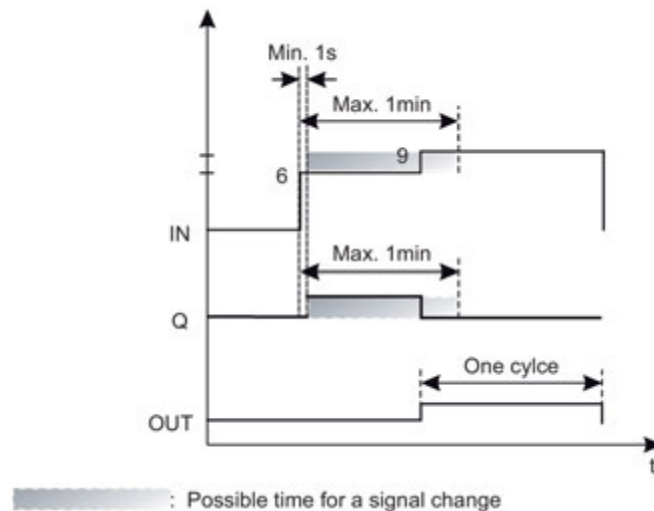
Inputs/outputs

| | Name | Data type | Description | Default |
|-----------------|------|-----------|-------------------------------|---------|
| Input: | IN | INT | Input | 0 |
| Outputs: | OUT | F_BOOL | ACKNOWLEDGMENT OUTPUT | 0 |
| | Q | BOOL | Status of the time evaluation | 0 |

Changing the collective signature of the offline safety program

If the above two acknowledgement steps are entered directly via the ES in CFC test mode rather than via the OS, the collective signature of the offline safety program changes as a result of the acknowledgment. To avoid this, you must ensure that a zero is entered after a 9 or an invalid value.

Timing Diagram



Operator control and monitoring

Parameters IN and Q have the system attribute S7_m_c. They can therefore be directly operated and monitored from an operator interface system (OS).

Error handling

An F_STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:

- "Safety program: Error in safety data format in DB" (Event ID 16#75DA).

A.2.5.12 F_TI_FTI: Conversion from TIME to F_TIME

Function

This F-Block converts the TIME data type at the IN input to the corresponding F_TIME F-Data type at the OUT output.

This enables signals formed in the standard user program to be evaluated in the safety program following a validity check (using F-Block F_LIM_TI, for example).

Inputs/outputs

| | Name | Data type | Description | Default |
|---------|------|-----------|-------------|---------|
| Input: | IN | TIME | Input | T# 0ms |
| Output: | OUT | F_TIME | Output | T# 0ms |

Error handling

None

A.2.5.13 F_I_FI: Conversion from INT to F_INT

Function

This F-Block converts the INT data type at the IN input to the corresponding F_INT F-Data type at the OUT output.

This enables signals formed in the standard user program to be evaluated in the safety program following a validity check (using F-Block F_LIM_I, for example).

Inputs/outputs

| | Name | Data type | Description | Default |
|----------------|------|-----------|-------------|---------|
| Input: | IN | INT | Input | 0 |
| | | | | |
| Output: | OUT | F_INT | Output | 0 |

Error handling

None

A.2.5.14 F_FI_FR: Conversion from F_INT to F_REAL

Function

This F-Block converts the F_INT F-Data type at the IN input to the F_REAL F-Data type at the OUT output.

Inputs/outputs

| | Name | Data type | Description | Default |
|----------------|------|-----------|-------------|---------|
| Input: | IN | F_INT | Input | 0 |
| | | | | |
| Output: | OUT | F_REAL | Output | 0.0 |

Error handling

None

A.2.5.15 F_FR_FI: Conversion from F_REAL to F_INT

Function

This F-Block converts the F_REAL F-Data type at the IN input to the F_INT F-Data type at the OUT output.

If the value at the IN input exceeds the upper limit which can be portrayed by the INT data type (range: -32768 to +32767), +32767 is output at the OUT output and output OUTU is set to 1. If the value lower range is violated, -32768 is output and the OUTL output is set to 1.

Inputs/outputs

| | Name | Data type | Description | Default |
|----------------|------|-----------|------------------------------|---------|
| Input: | IN | F_REAL | Input | 0.0 |
| Output: | OUT | F_INT | Output | 0 |
| | OUTU | F_BOOL | Upper number range violation | 0 |
| | OUTL | F_BOOL | Lower number range violation | 0 |

Error handling

- If the IN input is an invalid floating point number (NaN), 0 is output at the OUT output and OUTU and OUTL are set to 1.
- An F-STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.5.16 F_CHG_R: Safety Data Write for F_REAL

Function

The F-block F_CHG_R enables changes to be made to F-parameters in the safety program of the F-CPU from an operator station (Safety Data Write). For this purpose, the F-block implements a special safety protocol and monitors the required operator sequence.

The F-block can only be used in conjunction with the associated faceplate in the OS (see "Connection to the faceplate" below).

The OUT output is interconnected in the safety program with the I/O whose value is to be changed.

The limits for the change are specified using the MIN and MAX inputs.

The maximum increment of the change is specified at the MAXDELTA input.

The time during which the change must be completed is specified at the TIMEOUT input.


If a change was made on the faceplate in the required operator sequence within the F-monitoring time assigned for the TIMEOUT input, the value entered on the faceplate is made available at the OUT output, provided it meets the following conditions:

- The value is within the limits assigned for the MIN and MAX inputs.
- The maximum increment of the change assigned for the MAXDELTA input is not exceeded.


The "Safety Data Write" functionality must be enabled using input EN_CHG = 1.

Note

If EN_CHG changes to 0 during a transaction that has started already, the final value that is confirmed by the Confirmer is made available at the OUT output only when the EN_CHG input changes back to 1 (within the F-monitoring time).

| |
|--|
|  WARNING |
| <p>The "Safety Data Write" functionality allows changes to the safety program to be made during RUN mode.</p> <p>As a result, the following additional safety measures are required:</p> <ul style="list-style-type: none">• Ensure that changes that may compromise plant safety cannot be made. You can use the EN_CHG input for this purpose, for example, by controlling it with a key-operated switch or on a process-specific basis via the safety program.• Ensure that only authorized persons can make changes. Examples:<ul style="list-style-type: none">– Control the EN_CHG input with a keyswitch.– Set up access protection for operator stations where the "Safety Data Write" function can be performed. <p>As an alternative to the measures above, select the MIN, MAX and MAXDELTA inputs in such a way that values that could compromise plant safety cannot be specified using Safety Data Write.</p> |


When input CS_MODE = 1, the value made available at the OUT output is applied to the CS_VAL input, and output CHANGED = 1 is set.

| |
|---|
|  WARNING |
| <p>The CHANGED output cannot be evaluated in the safety program.</p> <p>CHANGED = 1 merely indicates that a change at the OUT output has been transferred to the CS_VAL input.</p> <p>If the value changed using the "Safety Data Write" function is to be in effect after a cold restart, you must manually correct the value at the CS_VAL input in the offline program and load memory.</p> |

I/Os

| | Name | Data type | Explanation | Default |
|-----------------|----------|-----------|---|------------|
| Inputs: | SAFE_ID1 | F_DINT | Unique ID (Part 1) for interconnecting the block instance with the faceplate | 0 |
| | SAFE_ID2 | F_DINT | Unique ID (Part 2) for interconnecting the block instance with the faceplate | 0 |
| | TIMEOUT | F_TIME | Permissible time between initiation of an F-parameter change and completion of the transaction. | T#60000 ms |
| | MIN | F_REAL | Low limit for F-parameter change. | 0.0 |
| | MAX | F_REAL | High limit for F-parameter change | 100.0 |
| | MAXDELTA | F_REAL | Maximum change between the current value (OUT) and the new value. | 10.0 |
| | CS_VAL | F_REAL | Initial value for OUT at a cold restart | 0.0 |
| | CS_MODE | F_BOOL | 1 = Apply changed OUT to CS_VAL 0 = CS_VAL remains unchanged | 0 |
| | WS_MODE | F_BOOL | 1 = At a warm restart/RESTART for F_SHUTDOWN, the value at the OUT output is retained 0 = At a warm restart/RESTART for F_SHUTDOWN, OUT receives the value of CS_VAL | 1 |
| | EN_CHG | F_BOOL | Allows Safety Data Write to be enabled and disabled. 1 = Enable 0 = Disable | 0 |
| Outputs: | OUT | F_REAL | Current fail-safe REAL value that is being used by the F-program | 0.0 |
| | CHANGED | BOOL | 1 = CS_VAL was changed via the HMI. | 0 |
| | CS_USED | F_BOOL | Indicates the value that was made available for OUT after F-startup 1 = CS_VAL 0 = Last valid value | 0 |

| | Name | Data type | Explanation | Default |
|--|--------|-------------|---|---------|
| | DIAG | WORD | Diagnostic information Bit 0 = 1: Error in safety data format Bit 1 = 1: MIN error Bit 2 = 1: MAX error Bit 3 = 1: DELTA error Bit 4 = 1: TIMEOUT error Bit 5 = 1: ID1 error Bit 6 = 1: ID2 error Bit 7 = 1: ID1_C error Bit 8 = 1: ID2_C error Bit 9 = 1: Test_ID1 error Bit 10 = 1: Test_ID2 error Bit 11 = 1: Error in safety data format IN Bit 12 = 1: TIMEOUT error during OS test Bit 13 = 1: Error: Negative number at the TIMEOUT input Bits 14-15: Reserve | W#16#0 |
| | USER | STRING [24] | Login of current operator on the OS. | " |
| | CURR_R | REAL | Copy of OUT.DATA Here, the unit of measurement to be displayed in the faceplate can be assigned using the "Unit" I/O property. | 0.0 |

| |
|--|
|  WARNING |
| The MIN, MAX and MAXDELTA inputs must not be interconnected. |

Connection to the faceplate

Communication between a block instance and the assigned faceplate takes place in the background by means of a special safety protocol. To configure the communication relationship between a block instance and the assigned faceplate, select a pair of numbers (Part 1 and Part 2) that is unique from all others in the system. Assign the pair of numbers to the SAFE_ID1 and SAFE_ID2 parameters as follows:

- To the SAFE_ID1 and SAFE_ID2 inputs of F_CHG_R in your safety program in *CFC*
- To the SAFE_ID1 and SAFE_ID2 parameters of the associated block icon in the *WinCC Graphics Designer*

WARNING

Parameters SAFE_ID1 and SAFE_ID2

The pair of numbers SAFE_ID1 and SAFE_ID2 of an F-block instance must be unique from all others in the system.

An instance of the F-block and the block icon of the associated faceplate must be given the same pair of numbers for the SAFE_ID1 and SAFE_ID2 parameters.

The SAFE_ID1 parameter must be programmed not equal to 0 and unique from all others in the program.

Startup behavior

After an F-startup, the F-block behaves as follows:

- After a CPU STOP followed by a cold restart of the F-CPU or during initial run:

In the first cycle after a cold restart or during the initial run, the value assigned at the CS_VAL input is made available at the OUT output. The CS_USED output is set to 1. CS_USED is reset to 0 as soon as "Secure Write Command++" has been successfully completed for the first time.

Note


The configured value at the CS_VAL input must be between the MIN and MAX values.

- After a CPU STOP followed by a warm restart of the F-CPU or after an F-STOP followed by a positive edge at the RESTART input of the F-block F_SHUTDOWN:

In the first cycle after a warm restart or after a positive edge at the RESTART input of the F-block F_SHUTDOWN, the last valid value of OUT is made available at the OUT output when input WS_MODE = 1. The CS_USED output retains its default value (0). When input WS_MODE = 0, the F-block behaves the same as after a cold restart.

Note

Prior to initial processing of the F-block after an F-startup, the default value is at the OUT and CS_USED outputs.

| |
|---|
|  WARNING |
| <p>F-startup</p> <p>After an F-startup, plant safety must not be compromised due to either the presence of the CS_VAL value at the OUT output or the presence of the last valid value at the OUT output.</p> <p>If necessary, evaluate the CS_USED output to determine whether the CS_VAL value or the last valid value was made available at the OUT output after an F-startup. You are not permitted to change the default value "0" of CS_USED.</p> <p>If a restart (warm restart) is performed after a cold restart, CS_USED is reset to the default value (0), even if the CS_VAL value is currently present at output OUT.</p> |

Error handling

- An F_STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostics event is then entered in the diagnostics buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)
- The DIAG output of the F-block signals if an error has been detected. This output must be checked if a transaction (Safety Data Write) fails. The individual errors remain active until the failed action has been repeated successfully. The individual bits have the following meanings:

| | |
|--------------|--|
| All bits = 0 | No problem; error-free operation |
| Bit 0 = 1 | Error in the safety data format at an input of the F-block |
| Bit 1 = 1 | MIN error: Transaction failed because the changed value is less than the MIN limit. |
| Bit 2 = 1 | MAX error: Transaction failed because the changed value is greater than the MAX limit. |
| Bit 3 = 1 | DELTA error: Transaction failed because the increment of the change exceeds the permissible MAXDELTA value; the changed value must be between OUT ± MAXDELTA. |
| Bit 4 = 1 | TIMEOUT error: A transaction was initiated but not completed within the specified time. |
| Bit 5 = 1 | ID1 error: Transaction failed because SAFE_ID1 does not match at the F-block instance and the faceplate in the OS. |
| Bit 6 = 1 | ID2 error: Transaction failed because SAFE_ID2 does not match at the F-block instance and the faceplate in the OS. |
| Bit 7 = 1 | ID1_C error: Transaction failed because SAFE_ID1 does not match at the F-block instance and the faceplate in the OS. |
| Bit 8 = 1 | ID2_C error: Transaction failed because SAFE_ID2 does not match at the F-block instance and the faceplate in the OS. |
| Bit 9=1 | Test_ID1 error: OS test failed because SAFE_ID1 does not match at the F-block instance and the faceplate in the OS. |

| | |
|------------|---|
| Bit 10 = 1 | Test_ID2 error: OS test failed because SAFE_ID2 does not match at the F-block instance and the faceplate in the OS. |
| Bit 11 = 1 | Error in safety data format IN: Transaction failed because of an error in the safety data format of the new value of the faceplate |
| Bit 12 = 1 | TIMEOUT error: During OS test |
| Bit 13 = 1 | TIMEOUT error: Negative number at the TIMEOUT input of the F-block |

See also

Configuring the Faceplate for Safety Data Write. (Page 165)

A.2.5.17 F_CHG_BO: Safety Data Write for F_BOOL**Function**

The F-block F_CHG_BO enables changes to be made to F-parameters in the safety program of the F-CPU from an operator station (Safety Data Write). For this purpose, the F-block implements a special safety protocol and monitors the required operator sequence.

The F-block can only be used in conjunction with the associated faceplate in the OS (see "Connection to the faceplate" below).

The OUT output is interconnected in the safety program with the I/O whose value is to be changed.


The time during which the change must be completed is specified at the TIMEOUT input.

If a change was made on the faceplate in the required operator sequence within the F-monitoring time assigned for the TIMEOUT input, the value entered on the faceplate is made available at the OUT output.


The "Safety Data Write" functionality must be enabled using input EN_CHG = 1.

Note

If EN_CHG changes to 0 during a transaction that has started already, the final value that is confirmed by the Confirmer is made available at the OUT output only when the EN_CHG input changes back to 1 (within the F-monitoring time).

| |
|---|
|  WARNING |
| <p>The "Safety Data Write" functionality allows changes to the safety program to be made during RUN mode.</p> <p>As a result, the following additional safety measures are required:</p> <ul style="list-style-type: none"> • Ensure that changes that may compromise plant safety cannot be made. You can use the EN_CHG input for this purpose, for example, by controlling it with a key-operated switch or on a process-specific basis via the safety program. • Ensure that only authorized persons can make changes. Examples: <ul style="list-style-type: none"> – Control the EN_CHG input with a keyswitch. – Set up access protection for operator stations where the "Safety Data Write" function can be performed. |

When input CS_MODE = 1, the value made available at the OUT output is applied to the CS_VAL input, and output CHANGED = 1 is set.

| |
|--|
|  WARNING |
| <p>The CHANGED output cannot be evaluated in the safety program.</p> <p>CHANGED = 1 merely indicates that a change at the OUT output has been transferred to the CS_VAL input.</p> <p>If the value changed using the "Safety Data Write" function is to be in effect after a cold restart, you must manually correct the value at the CS_VAL input in the offline program and load memory.</p> |

I/Os


| | Name | Data type | Explanation | Default |
|---------|----------|-----------|---|------------|
| Inputs: | SAFE_ID1 | F_DINT | Unique ID (Part 1) for interconnecting the block instance with the faceplate | 0 |
| | SAFE_ID2 | F_DINT | Unique ID (Part 2) for interconnecting the block instance with the faceplate | 0 |
| | TIMEOUT | F_TIME | Permissible time between initiation of an F-parameter change and completion of the transaction. | T#60000 ms |
| | CS_VAL | F_BOOL | Initial value for OUT at a cold restart | 0 |
| | CS_MODE | F_BOOL | 1 = Apply changed OUT to CS_VAL 0 = CS_VAL remains unchanged | 0 |
| | WS_MODE | F_BOOL | 1 = At a warm restart/RESTART for F_SHUTDOWN, the value at the OUT output is retained 0 = At a warm restart/RESTART for F_SHUTDOWN, OUT receives the value of CS_VAL | 1 |

| | Name | Data type | Explanation | Default |
|-----------------|---------|----------------|---|---------|
| | EN_CHG | F_BOOL | Allows Safety Data Write to be enabled and disabled. 1 = Enable 0 = Disable | 0 |
| Outputs: | OUT | F_BOOL | Current fail-safe BOOL value that is being used by the safety program | 0 |
| | CHANGED | BOOL | 1 = CS_VAL was changed via the HMI. | 0 |
| | CS_USED | F_BOOL | Indicates the value that was made available for OUT after startup. 1 = CS_VAL 0 = Last valid value | 0 |
| | DIAG | WORD | Diagnostic information Bit 0 = 1: Error in safety data format Bit 1 = 1: Reserve Bit 2 = 1: Reserve Bit 3 = 1: Reserve Bit 4 = 1: TIMEOUT error Bit 5 = 1: ID1 error Bit 6 = 1: ID2 error Bit 7 = 1: ID1_C error Bit 8 = 1: ID2_C error Bit 9 = 1: Test_ID1 error Bit 10 = 1: Test_ID2 error Bit 11 = 1: Error in safety data format IN Bit 12 = 1: TIMEOUT error during OS test Bit 13 = 1: Error: Negative number at the TIMEOUT input Bits 14-15: Reserve | W#16#0 |
| | USER | STRING [24] | Login of current operator on the OS. | " |

Connection to the faceplate

Communication between a block instance and the assigned faceplate takes place in the background by means of a special safety protocol. To configure the communication relationship between a block instance and the assigned faceplate, select a pair of numbers (Part 1 and Part 2) that is unique from all others in the system. Assign the pair of numbers to the SAFE_ID1 and SAFE_ID2 parameters as follows:

- to the SAFE_ID1 and SAFE_ID2 inputs of F_CHG_BO in your safety program in *CFC*
- to the SAFE_ID1 and SAFE_ID2 parameters of the associated block icon in the *WinCC Graphics Designer*

| |
|--|
|  WARNING |
| Parameters SAFE_ID1 and SAFE_ID2 |
| The pair of numbers SAFE_ID1 and SAFE_ID2 of an F-block instance must be unique from all others in the system. |
| An instance of the F-block and the block icon of the associated faceplate must be given the same pair of numbers for the SAFE_ID1 and SAFE_ID2 parameters. |
| The SAFE_ID1 parameter must be programmed not equal to 0 and unique from all others in the program. |


Startup behavior

After an F-startup, the F-block behaves as follows:

- After a CPU STOP followed by a cold restart of the F-CPU or during initial run:
In the first cycle after a cold restart or during the initial run, the value assigned at the CS_VAL input is made available at the OUT output. The CS_USED output is set to 1. CS_USED is reset to 0 as soon as "Secure Write Command++" has been successfully completed for the first time.
- After a CPU STOP followed by a warm restart of the F-CPU or after an F-STOP followed by a positive edge at the RESTART input of the F-block F_SHUTDOWN:
In the first cycle after a warm restart or after a positive edge at the RESTART input of the F-block F_SHUTDOWN, the last valid value of OUT is made available at the OUT output when input WS_MODE = 1. The CS_USED output retains its default value (0). When input WS_MODE = 0, the F-block behaves the same as after a cold restart.

Note

Prior to the initial processing of the F-block after an F-startup, the default value is at the OUT and CS_USED outputs.

| |
|---|
|  WARNING |
| <p>F-startup</p> <p>After an F-startup, plant safety must not be compromised due to either the presence of the CS_VAL value at the OUT output or the presence of the last valid value at the OUT output.</p> <p>If necessary, evaluate the CS_USED output to determine whether the CS_VAL value or the last valid value was made available at the OUT output after an F-startup. You are not permitted to change the default value "0" of CS_USED.</p> <p>If a restart (warm restart) is performed after a cold restart, CS_USED is reset to the default value (0), even if the CS_VAL value is currently present at output OUT.</p> |

Error handling

- An F_STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostics event is then entered in the diagnostics buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)
- The DIAG output of the F-block signals if an error has been detected. This output must be checked if a transaction (Safety Data Write) fails. The individual errors remain active until the failed action has been repeated successfully. The individual bits have the following meanings:

| | |
|--------------|---|
| All bits = 0 | No problem; error-free operation |
| Bit 0 = 1 | Error in the safety data format at an input of the F-block |
| Bit 1 = 1 | Reserve |
| Bit 2 = 1 | Reserve |
| Bit 3 = 1 | Reserve |
| Bit 4 = 1 | TIMEOUT error: A transaction was initiated but not completed within the specified time. |
| Bit 5 = 1 | ID1 error: Transaction failed because SAFE_ID1 does not match at the F-block instance and the faceplate in the OS. |
| Bit 6 = 1 | ID2 error: Transaction failed because SAFE_ID2 does not match at the F-block instance and the faceplate in the OS. |
| Bit 7 = 1 | ID1_C error: Transaction failed because SAFE_ID1 does not match at the F-block instance and the faceplate in the OS. |
| Bit 8 = 1 | ID2_C error: Transaction failed because SAFE_ID2 does not match at the F-block instance and the faceplate in the OS. |
| Bit 9 = 1 | Test_ID1 error: OS test failed because SAFE_ID1 does not match at the F-block instance and the faceplate in the OS. |
| Bit 10 = 1 | Test_ID2 error: OS test failed because SAFE_ID2 does not match at the F-block instance and the faceplate in the OS. |

| | |
|------------|---|
| Bit 11 = 1 | Error in safety data format IN: Transaction failed because of an error in the safety data format of the new value of the faceplate |
| Bit 12 = 1 | TIMEOUT error: During OS test |
| Bit 13 = 1 | TIMEOUT error: Negative number at the TIMEOUT input of the F-block |

See also

Configuring the Faceplate for Safety Data Write. (Page 165)

A.2.5.18 F_FBO_BO: Conversion from F_BOOL to BOOL

Function

This block converts F-Data type F_BOOL at input IN to the elementary data type BOOL at output OUT.

This enables you to evaluate signals that were generated in the safety program in the standard user program, as well.

This block must be placed in the standard user program.

Inputs/outputs

| | Name | Data type | Description | Default |
|----------------|------|-----------|-------------|---------|
| Input: | IN | F_BOOL | Input | — |
| Output: | OUT | BOOL | Output | — |

Error handling

None

A.2.5.19 F_FR_R: Conversion from F_REAL to REAL

Function

This block converts F-Data type F_REAL at input IN to the elementary data type REAL at output OUT.

This enables you to evaluate signals that were generated in the safety program in the standard user program, as well.

This block must be placed in the standard user program.

Inputs/outputs

| | Name | Data type | Description | Default |
|----------------|------|-----------|-------------|---------|
| Input: | IN | F_REAL | Input | — |
| Output: | OUT | REAL | Output | — |

Error handling

None

A.2.5.20 F_FL_I: Conversion from F_INT to INT**Function**

This block converts F-Data type F_INT at input IN to the elementary data type INT at output OUT.

This enables you to evaluate signals that were generated in the safety program in the standard user program, as well.

This block must be placed in the standard user program.

Inputs/outputs

| | Name | Data type | Description | Default |
|----------------|------|-----------|-------------|---------|
| Input: | IN | F_INT | Input | — |
| Output: | OUT | INT | Output | — |

Error handling

None

A.2.5.21 F_FTl_Tl: Conversion from F_TIME to TIME**Function**

This block converts F-Data type F_TIME at input IN to the elementary data type TIME at output OUT.

This enables you to evaluate signals that were generated in the safety program in the standard user program, as well.

This block must be placed in the standard user program.

Inputs/outputs

| | Name | Data type | Description | Default |
|---------|------|-----------|-------------|---------|
| Input: | IN | F_TIME | Input | — |
| Output: | OUT | TIME | Output | — |

Error handling

None

A.2.5.22 SWC_CHG: Operator function for Change process values

Function

This block is a standard block that establishes the connection to the faceplate. In addition, it provides all values for the display or handling of the protocol to the block icon and faceplate on the OS and generates messages for PCS 7 using the ALARM_8P.

Depending on the operator function, a SWC_CHG must be placed and inserted in the plant hierarchy.

Note

When used with PCS 7, one PO license is used for each instance of the SWC_CHG block in the safety program.

The following ALARM_8P messages are generated by this block for the alarm system:

- End-of-operator-input status

Note

When assigning the block name, note that the following illegal characters will be automatically replaced by the "\$" character during the transfer to the OS:

Space ? * ' :

Avoid these characters because an operator input is otherwise not possible.

Note

The creation of F-block types based on the "Secure Write Command++" function is not supported.

I/Os

| | Name | Data type | Explanation | Default |
|---------|---------|-------------|--|---------|
| Inputs: | ADR_SWC | DWORD | Connection between protocol block and operator control block | 0 |
| | NOTE | STRING [32] | Faceplate title | — |
| | AKT_V_B | BOOL | Actual value for the changed BOOL value | 0 |
| | AKT_V_R | REAL | Actual value for the changed REAL value | 0.0 |

A.2.5.23 SWC_MOS: Command function for Maintenance Override

SWC_MOS

This block is a standard block that establishes the connection to the faceplate. In addition, it provides all values for the display or handling of the protocol to the block icon and faceplate on the OS and generates messages for *PCS 7* using the ALARM_8P.

Depending on the operator function, a SWC_MOS must be placed and inserted in the plant hierarchy.

With the SWC_MOS block, only operator control of a fail-safe value is possible.

Note

When used with *PCS 7*, one PO license is used for each instance of the SWC_MOS block in the safety program.

The following ALARM_8P messages are generated by this block for the alarm system:

- Prewarning message for expiration of bypass time
- End-of-operator-input status
- Bypass active/not active

Note

When assigning the block name, note that the following illegal characters will be automatically replaced by the \$ character during the transfer to the OS:

Space ? * ' :

Avoid these characters because an operator input is otherwise not possible.

Note

The creation of F-block types based on the "Secure Write Command++" function is not supported.

I/Os

| | Name | Data type | Explanation | Default |
|----------------|----------|---|--|---------|
| Inputs: | ADR_SWC | DWORD | Connection between protocol block and operator control block | 0 |
| | NOTE | STRING [32] | Faceplate title | — |
| | AKT_B1 | BOOL | Actual value of 1st parameter for OS | 0 |
| | VMOD_B1B | BOOL | Status of channel (BOOL) | 0 |
| | Q_B1B | BOOL | Process value | 0 |
| | VMOD_B1R | REAL | Status of channel (REAL) | 0.0 |
| | V_B1R | REAL | Process value | 0.0 |
| | AKT_B2 | BOOL | Actual value of 2nd parameter for OS | 0 |
| | VMOD_B2B | BOOL | Status of channel (BOOL) | 0 |
| | Q_B2B | BOOL | Process value | 0 |
| | VMOD_B2R | REAL | Status of channel (REAL) | 0.0 |
| | V_B2R | REAL | Process value | 0.0 |
| | AKT_B3 | BOOL | Actual value of 3rd parameter for OS | 0 |
| | VMOD_B3B | BOOL | Status of channel (BOOL) | 0 |
| | Q_B3B | BOOL | Status of channel (BOOL) | 0 |
| | VMOD_B3R | REAL | Status of channel (REAL) | 0.0 |
| | V_B3R | REAL | Process value | 0.0 |
| | AKT_TR | BOOL | Actual value of retrigger signal for OS | 0 |
| | T_WARN | TIME | Prewarning time for active bypass | 0 s |
| | AKT_V_B | BOOL | Actual value of BOOL fail-safe value for OS | 0 |
| AKT_V_R | REAL | Actual value of REAL fail-safe value for OS | 0.0 | |
| MODE | WORD | Mutual interlock | W#16#0 | |

A.2.5.24 SWC_QOS: Operator function for fail-safe acknowledgment

Function

This block is a standard block that establishes the connection to the faceplate. In addition, it provides all values for the display or handling of the protocol to the block icon and faceplate on the OS and generates messages for PCS 7 using the ALARM_8P.

Depending on the operator function, a SWC_QOS must be placed and inserted in the plant hierarchy.

Note

When used with PCS 7, one PO license is used for each instance of the SWC_QOS block in the safety program.

The following ALARM_8P messages are generated by this block for the alarm system:

- End-of-operator-input status

Note

When assigning the block name, note that the following illegal characters will be automatically replaced by the "\$" character during the transfer to the OS:

Space ? * ' :

Avoid these characters because an operator input is otherwise not possible.

Note

The creation of F-block types based on the "Secure Write Command++" function is not supported.

I/Os

| | Name | Data type | Explanation | Default |
|----------------|---------|-------------|--|---------|
| Inputs: | ADR_SWC | DWORD | Connection between protocol block and operator control block | 0 |
| | NOTE | STRING [32] | Faceplate title | — |
| | AKT_Q | BOOL | Actual value for the acknowledgement | 0 |
| | ACK_REQ | BOOL | Actual value of the interconnected ACK_REQ | 0 |

A.2.6 F-Channel drivers for F-I/O

Overview

| Block name | Block number | Description |
|------------|--------------|---|
| F_CH_BI | FB 354 | F-channel driver for inputs of data type BOOL of fail-safe DP standard slaves and fail-safe IO standard devices |
| F_CH_BO | FB 355 | F-channel driver for outputs of data type BOOL of fail-safe DP standard slaves and fail-safe IO standard devices |
| F_PA_AI | FB 356 | F-channel driver for fail-safe PA field device "Transmitter" |
| F_PA_DI | FB 357 | F-channel driver for fail-safe PA field device "Discrete Input" |
| F_CH_DI | FB 377 | F-channel driver for digital inputs of F-I/O (except fail-safe DP standard slaves and fail-safe IO standard devices) |
| F_CH_DO | FB 378 | F-channel driver for digital outputs of F-I/O (except fail-safe DP standard slaves and fail-safe IO standard devices) |
| F_CH_AI | FB 379 | F-channel driver for analog inputs of F-I/O (except fail-safe DP standard slaves and fail-safe IO standard devices) |
| F_CH_II | FB 454 | F-channel driver for inputs of data type INT of fail-safe DP standard slaves and fail-safe IO standard devices |
| F_CH_IO | FB 455 | F-channel driver for outputs of data type INT of fail-safe DP standard slaves and fail-safe IO standard devices |
| F_CH_DII | FB 465 | F-channel driver for inputs of data type DINT of fail-safe DP standard slaves and fail-safe IO standard devices |
| F_CH_DIO | FB 466 | F-channel driver for outputs of data type DINT of fail-safe DP standard slaves and fail-safe IO standard devices |
| F_CH_RI | FB 616 | F-channel driver for inputs of data type REAL of fail-safe DP standard slaves and fail-safe IO standard devices |

A.2.6.1 F_CH_BI: F-Channel driver for inputs of data type BOOL of fail-safe DP standard slaves and fail-safe standard I/O devices

Function

The F-block is used for signal processing of an input value of data type BOOL of fail-safe DP standard slaves and fail-safe IO standard devices.

The F-block cyclically reads the input value addressed at the VALUE input of data type BOOL of a fail-safe DP standard slave/IO standard device from the associated F-module driver F_PS_12 that communicates with the fail-safe DP standard slave/IO standard device via a safety message frame according to the PROFIsafe bus profile. The F-module driver is automatically placed and interconnected with the CFC function "Generate module drivers".

If the digital input value is valid, it is made available at the Q output.

A quality code (QUALITY output) that can have the following states is generated for the result value at the Q output:

| State | Quality code (QUALITY output) |
|------------------------|-------------------------------|
| Valid value | 16#80 |
| Simulation | 16#60 |
| Fail-safe value | 16#48 |
| Invalid value (F-STOP) | 16#00 |

I/Os

| | Name | Data type | Explanation | Default |
|-----------------|----------|-----------|---|--------------------------------|
| Inputs: | ADR_CODE | DWORD | Coding for VALUE interconnection | to be automatically supplied * |
| | VALUE | BOOL | Address of the digital input channel | 0 |
| | SIM_I | F_BOOL | Simulation value | 0 |
| | SIM_ON | F_BOOL | 1 = Activate simulation value | 0 |
| | PASS_ON | F_BOOL | 1 = Activate passivation | 0 |
| | ACK_NEC | F_BOOL | 1 = User acknowledgement for reintegration after error required | 0 |
| | ACK_REI | F_BOOL | Acknowledgment for reintegration | 0 |
| | IPAR_EN | F_BOOL | 1 = Enable assignment of I-parameters | 0 |
| Outputs: | PASS_OUT | F_BOOL | 1 = Passivation due to error | 0 |
| | QBAD | F_BOOL | 1 = Process value invalid | 0 |
| | QSIM | F_BOOL | 1 = Simulation active | 0 |
| | Q | F_BOOL | Process value | 0 |
| | QN | F_BOOL | Process value inverted | 1 |
| | Q_DATA | BOOL | DATA component of the process value (for visualization) | 0 |
| | QUALITY | BYTE | Value status (quality code) of the process value | 0 |
| | Q_MOD | BOOL | Value from fail-safe DP standard slave/IO standard device | 0 |
| | ACK_REQ | BOOL | Acknowledgment for reintegration required | 0 |
| | IPAR_OK | F_BOOL | 1= New I-parameter values were assigned | 0 |

*) The ADR_CODE input is automatically supplied when the S7 program is compiled and must not be changed. The ADR_CODE input is displayed as changed during the comparison of safety programs if changes were made to the address or the symbol of the signal at the VALUE input.

Note

Forcing the VALUE input

Forcing of the VALUE input is not possible because VALUE is not the value itself but rather a pointer for the address of the digital input channel.

Addressing

You must interconnect the symbol generated with *HW Config* in the symbol table for the input value of data type BOOL with the VALUE input.

Note

An inversion of the VALUE input in the *CFC Editor* is ineffective. Use the QN output instead.

Normal value

If the input value received from the fail-safe DP standard slave/IO standard device is valid, it is output at the Q output with quality code (QUALITY) 16#80.

Simulation

A simulation value can be output at the Q output instead of the normal value that is received from the fail-safe DP standard slave/IO standard device.

When input SIM_ON = 1, the value of the SIM_I input with quality code (QUALITY) 16#60 is output. Simulation has the highest priority. QBAD is always set to "0". QSIM = 1 is set if the block is in simulation state.

When simulation is switched on, the input value received from the fail-safe DP standard slave/IO standard device is output at the Q_MOD output. If no communication is possible with the fail-safe DP standard slave/IO standard device or if there is still no user acknowledgement after an error, 0 is output.

When simulation is switched off, Q_DATA is output.

Fail-safe value

The fail-safe value 0 is output at the Q output in the following cases:

- The digital input value is invalid due to a communication error (PROFIsafe).
- The digital input value is invalid due to a module fault or a fail-safe value is received from the module.
- A passivation with PASS_ON = 1 is present.
- An F-startup is present.
- FV_ACTIVATED is signaled by the module.

The quality code (QUALITY) is set to 16#48 and QBAD = 1 is set.

If the output of the fail-safe value is not caused by a passivation, PASS_OUT = 1 is additionally set for passivation of other channels.

Parameter reassignment of a fail-safe DP standard slave/IO standard device

For parameter reassignment of a fail-safe DP standard slave/IO standard device, the IPAR_EN input and the IPAR_OK output are available.

The IPAR_EN input corresponds to the iPar_EN_C variable and the IPAR_OK output corresponds to the iPar_OK_S variable in the PROFIsafe bus profile as of PROFIsafe Specification V1.30. To find out when you must set/reset the IPAR_EN input following parameter reassignment of a fail-safe DP standard slave/IO standard device and how you can evaluate the IPAR_OK output, refer to the PROFIsafe Specification V1.30 or higher and the documentation for the fail-safe standard DP slave/IO standard device.

If more than one F-channel driver is placed for a fail-safe DP standard slave/IO standard device, iPar_EN_C is formed from an OR logic operation of all IPAR_EN of the F-channel driver belonging to the fail-safe DP standard slave/IO standard device.

If passivation is to be carried out when IPAR_EN = 1, you must additionally set variable PASS_ON = 1.

Reintegration after error elimination

After elimination of an error, the input value received from the fail-safe DP standard slave/IO standard device can be reintegrated automatically or only after user acknowledgment.

With parameter assignment ACK_NEC = 1, a user acknowledgement is required at the ACK_REI input after error elimination. With parameter assignment ACK_NEC = 0, reintegration occurs automatically.

Output ACK_REQ = 1 is used to signal that the error is eliminated and a user acknowledgment at the ACK_REI input is required for reintegration.

No user acknowledgement is required for reintegration after PASS_ON = 1. For reintegration after an F-startup following a CPU STOP, no user acknowledgement is required if the fail-safe DP standard slave/IO standard device starts up using the Slave State (20) "system start" according to the PROFIsafe Specification V1.30 and higher. Otherwise, a communication error (PROFIsafe) is detected.

WARNING

Parameter assignment of input ACK_NEC = 0 is only allowed if an automatic reintegration is permissible under safety aspects for the process.

Communication errors (PROFIsafe) must always be acknowledged at the ACK_REI input independent of ACK_NEC. For this purpose, you must interconnect the ACK_REI input with a signal generated by an operator input. An interconnection with an automatically generated signal is not permissible.

**WARNING****Startup protection for short-term power failure of the fail-safe DP standard slave/IO standard device**

After a power failure of the fail-safe DP standard slave/IO standard device that lasts for less time than the F-monitoring time (see section "Run times, F-Monitoring times, and response times (Page 460)") set for the fail-safe DP standard slaves/IO standard devices in *HW Config*, an automatic reintegration, as described for parameter assignment `ACK_NEC = 0`, may occur irrespective of your parameter assignment of the `ACK_NEC` input.

If automatic reintegration is not permitted for the process involved, you must program a startup protection for this situation by evaluating the `QBAD` or `PASS_OUT` outputs.

At a power failure of the fail-safe DP standard slave/IO standard device lasting longer than the F-monitoring time set for the fail-safe DP standard slave/IO standard device in *HW Config*, the F-system detects a communication error.

Startup behavior

After an F-startup, communication must first be established between the F-module driver and the fail-safe DP standard slave/IO standard device. During this time, the substitute value 0 with quality code (QUALITY output) 16#48 is output and outputs `QBAD = 1` and `PASS_OUT = 1` are additionally set.

Error handling

An F-STOP is triggered when there is an error in the safety data format in the associated instance DB. The following diagnostics event is then entered in the diagnostics buffer of the F-CPU:

- "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

Response to F-STOP

Following an F-STOP, 16#00 is output at the QUALITY output and `QBAD.DATA = 1` is set. All other variables are frozen.

A.2.6.2 F_CH_BO: F-Channel driver for outputs of data type BOOL of fail-safe DP standard slaves and fail-safe standard I/O devices**Function**

The F-block is used for signal processing of an output value of data type BOOL of fail-safe DP standard slaves and fail-safe IO standard devices.

The F-block cyclically writes the output value of data type BOOL for the output of a fail-safe DP standard slave/IO standard device addressed at the VALUE output to the associated F-module driver `F_PS_12` that communicates with the fail-safe DP standard slave/IO standard device via a safety message frame according to the PROFIsafe bus profile. The F-module

driver is automatically placed and interconnected with the *CFC* function "Generate module drivers".

A quality code, which can take the following states, is generated for the output value that is written to the fail-safe DP standard slave/IO standard device:

| State | Quality code |
|------------------------|--------------|
| Valid value | 16#80 |
| Simulation | 16#60 |
| Fail-safe value | 16#48 |
| Invalid value (F-STOP) | 16#00 |

I/Os

| | Name | Data type | Explanation | Default |
|-----------------|----------|-----------|---|--------------------------------|
| Inputs: | ADR_CODE | DWORD | Coding for VALUE interconnection | to be automatically supplied * |
| | I | F_BOOL | Process value | 0 |
| | SIM_I | F_BOOL | Simulation value | 0 |
| | SIM_MOD | F_BOOL | 1 = Simulation has priority | 0 |
| | SIM_ON | F_BOOL | 1 = Activate simulation value | 0 |
| | PASS_ON | F_BOOL | 1 = Activate passivation | 0 |
| | ACK_NEC | F_BOOL | 1 = User acknowledgement for reintegration after error required | 0 |
| | ACK_REI | F_BOOL | Acknowledgment for reintegration | 0 |
| | IPAR_EN | F_BOOL | 1 = Enable assignment of I-parameters | 0 |
| Outputs: | PASS_OUT | F_BOOL | 1 = Passivation due to error | 0 |
| | QBAD | F_BOOL | 1 = Process value invalid | 0 |
| | QSIM | F_BOOL | 1 = Simulation active | 0 |
| | VALUE | BOOL | Address of the digital output channel | 0 |
| | QUALITY | BYTE | Value status (quality code) of the process value | 0 |
| | ACK_REQ | BOOL | Acknowledgment for reintegration required | 0 |
| | IPAR_OK | F_BOOL | 1= New I-parameter values were assigned | 0 |

*) The ADR_CODE input is automatically supplied when the S7 program is compiled and must not be changed. The ADR_CODE input is displayed as changed during the comparison of safety programs if changes were made to the address or the symbol of the signal at the VALUE output.

Addressing

You must interconnect the symbol generated with *HW Config* in the symbol table for the output value of data type BOOL with the VALUE output.

Normal value

The process value active at the I input is written to the fail-safe DP standard slave/IO standard device. The quality code (QUALITY) is set to 16#80.

Simulation

A simulation value can also be written to the fail-safe DP standard slave/IO standard device instead of the process value active at the I input.

When input SIM_ON = 1 and input SIM_MOD = 0, the value of the SIM_I input is written to the fail-safe DP standard slave/IO standard device and output at the VALUE output, if no communication error (PROFIsafe), no module or channel fault (e.g. wire break) and no F-startup is present. The quality code (QUALITY) is set to 16#60.

When input SIM_ON = 1 and input SIM_MOD = 1, the value of the SIM_I input is output at the VALUE output even when a communication error (PROFIsafe), module or channel fault (e.g. wire break) or F-startup is present, in order to simulate an "error-free" operation without the presence of a real fail-safe DP standard slave/IO standard device.

In both cases, the quality code (QUALITY) is set to 16#60 and QSIM = 1 is set.

Note

If you have placed more than one F-channel driver for outputs for a fail-safe DP standard slave/IO standard device, a simulation value is not written if the PASS_ON input of another F-channel driver for outputs of the fail-safe DP standard slave/IO standard device is "1" and the SIM_ON input is 0.

Fail-safe value

The fail-safe value 0 is written to the fail-safe DP standard slave/IO standard device when any of the following occurs:

- A communication error (PROFIsafe)
- A module or channel fault (e.g. wire break)
- An F-startup
- Passivation with PASS_ON = 1

The quality code (QUALITY) is set to 16#48 and QBAD = 1 is set.

If the output of the fail-safe value is not caused by a passivation, PASS_OUT = 1 is additionally set for passivation of other channels.

Note

For fail-safe DP standard slaves/IO standard devices, channel-specific passivation is not possible for outputs using PASS_ON! If you have placed more than one F-channel driver for outputs for a fail-safe DP standard slave/IO standard device, the fail-safe value 0 is written for all outputs of the fail-safe DP standard slave/IO standard device following a passivation with PASS_ON = 1 at one of the F-channel drivers. If you want to evaluate the QBAD and QUALITY outputs of the other F-channel drivers when PASS_ON = 1 at one of the F-channel drivers, you must control the PASS_ON inputs of all F-channel drivers synchronously!

Parameter reassignment of a fail-safe DP standard slave/IO standard device

For parameter reassignment of a fail-safe DP standard slave/IO standard device, the IPAR_EN input and the IPAR_OK output are available.

The IPAR_EN input corresponds to the iPar_EN_C variable and the IPAR_OK output corresponds to the iPar_OK_S variable in the PROFIsafe bus profile as of PROFIsafe Specification V1.30. To find out when you must set/reset the IPAR_EN input following parameter reassignment of a fail-safe DP standard slave/IO standard device and how you can evaluate the IPAR_OK output, refer to the PROFIsafe Specification V1.30 or higher and the documentation for the fail-safe standard DP slave/IO standard device.

If more than one F-channel driver is placed for a fail-safe DP standard slave/IO standard device, iPar_EN_C is formed from an OR logic operation of all IPAR_EN of the F-channel driver belonging to the fail-safe DP standard slave/IO standard device.

If passivation is to be carried out when IPAR_EN = 1, you must additionally set variable PASS_ON = 1.

Reintegration after error elimination

After elimination of an error, the fail-safe DP standard slave/IO standard device can be reintegrated automatically or only after user acknowledgment. With parameter assignment ACK_NEC = 1, a user acknowledgment is required at the ACK_REI input after error elimination. With parameter assignment ACK_NEC = 0, reintegration occurs automatically.

Output ACK_REQ = 1 is used to signal that the error is eliminated and a user acknowledgment at the ACK_REI input is required for reintegration.

No user acknowledgement is required for reintegration after `PASS_ON = 1`. For reintegration after an F-startup following a CPU STOP, no user acknowledgment is required if the fail-safe DP standard slave/IO standard device starts up using the Slave State (20) "system start" according to the PROFIsafe Specification V1.30 and higher. Otherwise, a communication error (PROFIsafe) is detected.

Note

For fail-safe DP standard slaves/IO standard devices, channel-specific reintegration is not possible for outputs using `PASS_ON`! If you have placed more than one F-channel driver for outputs for a fail-safe DP standard slave/IO standard device, you must synchronously control the `ACK_REI` inputs of all F-channel drivers for outputs of the fail-safe DP standard slave/IO standard device.

WARNING

Parameter assignment of input `ACK_NEC = 0` is only allowed if an automatic reintegration is permissible under safety aspects for the process.

Communication errors (PROFIsafe) must always be acknowledged at the `ACK_REI` input independent of `ACK_NEC`. For this purpose, you must interconnect the `ACK_REI` input with a signal generated by an operator input. An interconnection with an automatically generated signal is not permissible.

WARNING

Startup protection for short-term power failure of the fail-safe DP standard slave/IO standard device

After a power failure of the fail-safe DP standard slave/IO standard device that lasts for less time than the F-monitoring time (see section "Run times, F-Monitoring times, and response times (Page 460)") set for the fail-safe DP standard slaves/IO standard devices in *HW Config*, an automatic reintegration, as described for parameter assignment `ACK_NEC = 0`, may occur irrespective of your parameter assignment of the `ACK_NEC` input.

If automatic reintegration is not permitted for the process involved, you must program a startup protection for this situation by evaluating the `QBAD` or `PASS_OUT` outputs.

At a power failure of the fail-safe DP standard slave/IO standard device lasting longer than the F-monitoring time set for the fail-safe DP standard slave/IO standard device in *HW Config*, the F-system detects a communication error.

Startup behavior

After an F-startup, communication must first be established between the F-module driver and the fail-safe DP standard slave/IO standard device. During this time, the fail-safe value 0 is written to the fail-safe DP standard slave/IO standard device. The quality code (QUALITY) is set to 16#48 and outputs `QBAD = 1` and `PASS_OUT = 1` are set.

Error handling

An F-STOP is triggered when there is an error in the safety data format in the associated instance DB. The following diagnostics event is then entered in the diagnostics buffer of the F-CPU:

- "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

Response to F-STOP

Following an F-STOP, 16#00 is output at the QUALITY output and QBAD.DATA = 1 is set. All other variables are frozen.

A.2.6.3 F_PA_AI: Fail-safe channel driver for fail-safe "Transmitter" PA field device

Function

The block is used for signal processing of an analog input value from a fail-safe slot (F-slot) of a "Transmitter" fail-safe PA field device.

The F-block cyclically reads the process value addressed at the VALUE input with status byte (quality code) of the fail-safe PA field device from the associated F-module driver that communicates with the F-slot of a fail-safe PA field device via a safety message frame according to the PROFIsafe bus profile. The F-module driver is automatically placed and interconnected with the *CFC* function "Generate module drivers".

If the process value exists as a physical quantity, it is made available at the V output. The status byte (quality code) is made available at the STATUS output and contains information about the status of the fail-safe PA field device.

A quality code (QUALITY output) that can have the following states is generated for the result value at the V output:

| State | Quality code (QUALITY output) |
|--|-------------------------------|
| Valid value | 16#80 |
| Simulation | 16#60 |
| Fail-safe value | 16#48 |
| Last valid value | 16#44 |
| Uncertain, device-related | 16#68 |
| Uncertain, process-related | 16#78 |
| Uncertain, device-related, range violation | 16#54 |
| Maintenance demanded | 16#A4 |
| Invalid value (F-STOP) | 16#00 |

I/Os

| | Name | Data type | Explanation | Default |
|-----------------|----------|-----------|---|-------------------------------|
| Inputs: | ADR_CODE | DWORD | Coding for VALUE interconnection | to be automatically supplied* |
| | VALUE | REAL | Address of the analog input channel | 0 |
| | SIM_V | F_REAL | Simulation value | 0.0 |
| | SIM_ON | F_BOOL | 1 = Activate simulation value | 0 |
| | SUBS_V | F_REAL | Fail-safe value | 0.0 |
| | SUBS_ON | F_BOOL | 1 = Enable fail-safe value injection | 0 |
| | PASS_ON | F_BOOL | 1 = Activate passivation | 0 |
| | ACK_NEC | F_BOOL | 1 = User acknowledgement for reintegration after error required | 0 |
| | ACK_REI | F_BOOL | Acknowledgment for reintegration | 0 |
| | IPAR_EN | F_BOOL | 1 = Enable assignment of I-parameters | 0 |
| Outputs: | PASS_OUT | F_BOOL | 1 = Passivation due to error | 0 |
| | QBAD | F_BOOL | 1 = Process value invalid | 0 |
| | QSIM | F_BOOL | 1 = Simulation active | 0 |
| | QSUBS | F_BOOL | 1 = Fail-safe value injection active | 0 |
| | V | F_REAL | Process value | 0.0 |
| | V_DATA | REAL | DATA component of the process value (for visualization) | 0.0 |
| | QUALITY | BYTE | Value status (quality code) of the process value | 0 |
| | STATUS | BYTE | Process value status | 0 |
| | V_MOD | REAL | Value of F-PA field device | 0.0 |
| | ACK_REQ | BOOL | Acknowledgment for reintegration required | 0 |
| | IPAR_OK | F_BOOL | 1= New I-parameter values were assigned | 0 |

*) The ADR_CODE input is automatically supplied when the S7 program is compiled and must not be changed. The ADR_CODE input is displayed as changed during the comparison of safety programs if changes were made to the address or the symbol of the signal at the VALUE input.

Addressing

You must interconnect the symbol generated with *HW Config* in the symbol table for the analog input channel with the VALUE input.

Normal value

If the analog input value received from the fail-safe PA field device is valid, it is output at the V output. The quality code (QUALITY) is set to 16#80, 16#54, 16#60, 16#68, 16#78 or 16#A4 depending on the quality code received from the fail-safe PA field device.

Simulation

A simulation value can be output at the V output instead of the normal value that is received from the fail-safe PA field device.

When input SIM_ON = 1, the value of the SIM_V input with quality code (QUALITY) 16#60 is output. Simulation has the highest priority. QBAD and QSUBS are always set = 0. If the block is in simulation state due to SIM_ON = 1, then QSIM = 1 is set.

Note

Quality code (QUALITY) 16#60 is also output, if a simulation was started on the fail-safe PA field device and there is no event for the output of a fail-safe value or last valid value.

When simulation is switched on, the input value received from the fail-safe PA field device is output at the V_MOD output. If no communication is possible with the fail-safe PA field device or if there is still no user acknowledgement after an error, 0.0 is output.

When simulation is switched off, V_DATA is output.

Fail-safe value

When input SUBS_ON = 1, the fail-safe value SUBS_V is output at the V output in the following cases:

- The analog input value is invalid due to a communication error (PROFIsafe).
- The analog input value is invalid due to a module fault or a fail-safe value is received from the module.
- A passivation with PASS_ON = 1 is present.
- An F-startup is present.
- FV_ACTIVATED is signaled by the module.

The quality code (QUALITY) is set to 16#48 and QSUBS = 1 and QBAD = 1 are set.

If the output of the fail-safe value is not caused by a passivation, PASS_OUT = 1 is additionally set for passivation of other channels.

Keep last value

When input SUBS_ON = 0, the last valid value of V is output at the V output in the following cases:

- The analog input value is invalid due to a communication error (PROFIsafe).
- The analog input value is invalid due to a module fault or a fail-safe value is received from the module.
- A passivation with PASS_ON = 1 is present.
- FV_ACTIVATED is signaled by the module.

The quality code (QUALITY) is set to 16#44 and QSUBS = 0 and QBAD = 1 are set.

If the output of the last valid value is not caused by a passivation, then output PASS_OUT = 1 is additionally set in order to passivate other channels.

Parameter reassignment of a fail-safe PA field device

For parameter reassignment of a fail-safe PA field device, the IPAR_EN input and the IPAR_OK output are available.

The IPAR_EN input corresponds to the IPar_EN_C variable and the IPAR_OK output corresponds to the IPar_OK_S variable in the PROFIsafe bus profile as of PROFIsafe Specification V1.30. To find out when you must set/reset the IPAR_EN input following parameter reassignment of a fail-safe PA field device and how you can evaluate the IPAR_OK output, refer to the PROFIsafe Specification V1.30 or higher and the documentation for the fail-safe PA field device.

If more than one F-channel driver is placed for an F-slot of a fail-safe PA field device, IPar_EN_C is formed from an OR logic operation of all IPAR_EN of the F-channel drivers belonging to the F-slot of the fail-safe PA field device.

If passivation is to be carried out when IPAR_EN = 1, you must additionally set variable PASS_ON = 1.


Reintegration after error elimination

After elimination of an error, the analog input value received from the fail-safe PA field device can be reintegrated automatically or only after user acknowledgment.

With parameter assignment ACK_NEC = 1, a user acknowledgement is required at the ACK_REI input after error elimination. With parameter assignment ACK_NEC = 0, reintegration occurs automatically.

Output ACK_REQ = 1 is used to signal that the error is eliminated and a user acknowledgment at the ACK_REI input is required for reintegration.

No user acknowledgement is required for reintegration after PASS_ON = 1. For reintegration after an F-startup following a CPU STOP, no user acknowledgment is required if the fail-safe PA field device starts up using the Slave State (20) "system start" according to the PROFIsafe Specification V1.30 and higher. Otherwise, a communication error (PROFIsafe) is detected.

| |
|---|
|  WARNING |
| Parameter assignment of input ACK_NEC = 0 is only allowed if an automatic reintegration is permissible under safety aspects for the process. |
| Communication errors (PROFIsafe) must always be acknowledged at the ACK_REI input independent of ACK_NEC. For this purpose, you must interconnect the ACK_REI input with a signal generated by an operator input. An interconnection with an automatically generated signal is not permissible. |

⚠ WARNING**Startup protection for short-term power failure of the fail-safe PA field device**

After a power failure of the fail-safe PA field device that lasts for less time than the F-monitoring time (see section "Run times, F-Monitoring times, and response times (Page 460)") set for the fail-safe PA field device in *HW Config*, an automatic reintegration, as described for when ACK_NEC = 0 is configured, may occur irrespective of your parameter assignment of the ACK_NEC input.

If automatic reintegration is not permitted for the process involved, you must program a startup protection for this situation by evaluating the QBAD or PASS_OUT outputs.

At a power failure of the fail-safe PA field device lasting longer than the F-monitoring time set for the fail-safe PA field device in *HW Config*, the F-system detects a communication error.

Startup behavior

After an F-startup, communication must first be established between the F-module driver and the fail-safe PA field device. During this time, the fail-safe value SUBS_V with quality code (QUALITY output) 16#48 is output irrespective of the parameter assignment at the SUBS_ON input and outputs QSUBS = 1, QBAD = 1 and PASS_OUT = 1 are additionally set.

Error handling

An F-STOP is triggered when there is an error in the safety data format in the associated instance DB. The following diagnostics event is then entered in the diagnostics buffer of the F-CPU:

- "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

Response to F-STOP

Following an F-STOP, 16#00 is output at the QUALITY and STATUS outputs and QBAD.DATA = 1 is set. All other variables are frozen.

A.2.6.4 F_PA_DI: Fail-safe channel driver for fail-safe "Discrete Input" PA field device**Function**

The F-block is used for signal processing of a digital input value from a fail-safe slot (F-slot) of a "Discrete Input" fail-safe PA field device.

The F-block cyclically reads the process value addressed at the I_OUT_D input with status byte (quality code) of the fail-safe PA field device from the associated F-module drive that communicates with the F-slot of a fail-safe PA field device via a safety message frame according to the PROFIsafe bus profile. The F-module driver is automatically placed and interconnected with the *CFC* function "Generate module drivers".

If the process value is valid, the bit (0 to 7) assigned at the BIT_NR input is made available by the process value (byte) at the Q output. The status byte (quality code) is made available at the STATUS output and contains information about the status of the fail-safe PA field device.

A quality code (QUALITY output) that can have the following states is generated for the result value at the Q output:

| State | Quality code (QUALITY output) |
|--|-------------------------------|
| Valid value | 16#80 |
| Simulation | 16#60 |
| Fail-safe value | 16#48 |
| Uncertain, device-related | 16#68 |
| Uncertain, process-related | 16#78 |
| Uncertain, device-related, range violation | 16#54 |
| Maintenance demanded | 16#A4 |
| Invalid value (F-STOP) | 16#00 |

I/Os

| | Name | Data type | Explanation | Default |
|-----------------|----------|-----------|---|-------------------------------|
| Inputs: | ADR_CODE | DWORD | Coding for I_OUT_D interconnection | to be automatically supplied* |
| | BIT_NR | F_INT | REQUIRED BIT NUMBER 0 ... 7 | 0 |
| | I_OUT_D | BYTE | Address of the process value | 0 |
| | SIM_I | F_BOOL | Simulation value | 0 |
| | SIM_ON | F_BOOL | 1 = Activate simulation value | 0 |
| | PASS_ON | F_BOOL | 1 = Activate passivation | 0 |
| | ACK_NEC | F_BOOL | 1 = User acknowledgement for reintegration after error required | 0 |
| | ACK_REI | F_BOOL | Acknowledgment for reintegration | 0 |
| | IPAR_EN | F_BOOL | 1 = Enable assignment of I-parameters | 0 |
| Outputs: | PASS_OUT | F_BOOL | 1 = Passivation due to error | 0 |
| | QBAD | F_BOOL | 1 = Process value invalid | 0 |
| | QSIM | F_BOOL | 1 = Simulation active | 0 |
| | Q | F_BOOL | Process value | 0 |
| | QN | F_BOOL | Process value inverted | 1 |
| | Q_DATA | BOOL | DATA component of the process value (for visualization) | 0 |
| | QUALITY | BYTE | Value status (quality code) of the process value | 0 |
| | STATUS | BYTE | Status of the process value | 0 |
| | Q_MOD | BOOL | Value from the fail-safe PA field device | 0 |
| | Q0 | BOOL | Process value bit 0 | 0 |
| | ... | | ... | |

| | Name | Data type | Explanation | Default |
|--|---------|-----------|---|---------|
| | Q7 | BOOL | Process value bit 7 | 0 |
| | ACK_REQ | BOOL | Acknowledgment for reintegration required | 0 |
| | IPAR_OK | F_BOOL | 1= New I-parameter values were assigned | 0 |

*) The ADR_CODE input is automatically supplied when the S7 program is compiled and must not be changed. The ADR_CODE input is displayed as changed during the comparison of safety programs if changes were made to the address or the symbol of the signal at the I_OUT_D input.

Addressing

You must interconnect the symbol generated with *HW Config* in the symbol table for the process value with the I_OUT_D input.

Note

If the symbol for the process value that was generated in the symbol table with *HW Config* was not generated with the data type "BYTE", but rather with the data type "BOOL", you must add a symbol with the data type BYTE yourself in the symbol table.

Normal value

If the digital input value received from the fail-safe PA field device is valid, it is output at the Q output. The quality code (QUALITY) is set to 16#80, 16#54, 16#60, 16#68, 16#78 or 16#A4 depending on the quality code received from the fail-safe PA field device.

Simulation

A simulation value can be output at the Q output instead of the normal value that is received from the fail-safe PA field device.

When input SIM_ON = 1, the value of the SIM_I input with quality code (QUALITY) 16#60 is output. Simulation has the highest priority. QBAD is always set to "0". If the F-block is in simulation state due to SIM_ON = 1, then QSIM = 1 is set.

Note

Quality code (QUALITY) 16#60 is also output, if a simulation was started on the fail-safe PA field device and there is no event for the output of a fail-safe value.

When simulation is switched on, the digital input value received from the fail-safe PA field device is output at the Q_MOD output. If no communication is possible with the fail-safe PA field device or if there is still no user acknowledgement after an error, 0 is output.

When simulation is switched off, Q_DATA is output.

Fail-safe value

The fail-safe value 0 is output at the Q output in the following cases:

- The digital input value is invalid due to a communication error (PROFIsafe).
- The digital input value is invalid due to a module fault or a fail-safe value is received from the module.
- A passivation with PASS_ON = 1 is present.
- An F-startup is present.
- FV_ACTIVATED is signaled by the module.

The quality code (QUALITY) is set to 16#48 and QBAD = 1 is set.

If the output of the fail-safe value is not caused by a passivation, PASS_OUT = 1 is additionally set for passivation of other channels.

Parameter reassignment of a fail-safe PA field device

For parameter reassignment of a fail-safe PA field device, the IPAR_EN input and the IPAR_OK output are available.

The IPAR_EN input corresponds to the iPar_EN_C variable and the IPAR_OK output corresponds to the iPar_OK_S variable in the PROFIsafe bus profile as of PROFIsafe Specification V1.30. To find out when you must set/reset the IPAR_EN input following parameter reassignment of a fail-safe PA field device and how you can evaluate the IPAR_OK output, refer to the PROFIsafe Specification V1.30 or higher and the documentation for the fail-safe PA field device.

If more than one F-channel driver is placed for an F-slot of a fail-safe PA field device, iPar_EN_C is formed from an OR logic operation of all IPAR_EN of the F-channel drivers belonging to the F-slot of the fail-safe PA field device.

If passivation is to be carried out when IPAR_EN = 1, you must additionally set variable PASS_ON = 1.

Reintegration after error elimination

After elimination of an error, the digital input value received from the fail-safe PA field device can be reintegrated automatically or only after user acknowledgment.

With parameter assignment ACK_NEC = 1, a user acknowledgement is required at the ACK_REI input after error elimination. With parameter assignment ACK_NEC = 0, reintegration occurs automatically.

Output ACK_REQ = 1 is used to signal that the error is eliminated and a user acknowledgment at the ACK_REI input is required for reintegration.

No user acknowledgement is required for reintegration after PASS_ON = 1. For reintegration after an F-startup following a CPU STOP, no user acknowledgment is required if the F-I/O starts up using the Slave State (20) "system start" according to the PROFIsafe Specification V1.30 and higher. Otherwise, a communication error (PROFIsafe) is detected.

⚠ WARNING

Parameter assignment of input ACK_NEC = 0 is only allowed if an automatic reintegration is permissible under safety aspects for the process.

Communication errors (PROFIsafe) must always be acknowledged at the ACK_REI input independent of ACK_NEC. For this purpose, you must interconnect the ACK_REI input with a signal generated by an operator input. An interconnection with an automatically generated signal is not permissible.

⚠ WARNING**Startup protection for short-term power failure of the fail-safe PA field device**

After a power failure of the fail-safe PA field device that lasts for less time than the F-monitoring time (see section "Run times, F-Monitoring times, and response times (Page 460)") set for the fail-safe PA field device in *HW Config*, an automatic reintegration, as described for when ACK_NEC = 0 is configured, may occur irrespective of your parameter assignment of the ACK_NEC input.

If automatic reintegration is not permitted for the process involved, you must program a startup protection for this situation by evaluating the QBAD or PASS_OUT outputs.

At a power failure of the fail-safe PA field device lasting longer than the F-monitoring time set for the fail-safe PA field device in *HW Config*, the F-system detects a communication error.

Startup behavior

After an F-startup, communication must first be established between the F-module driver and the fail-safe PA field device. During this time, the substitute value 0 with quality code (QUALITY output) 16#48 is output and outputs QBAD = 1 and PASS_OUT = 1 are additionally set.

Error handling

- If the BIT_NR input is assigned a value $\neq 0 \dots 7$, the fail-safe value 0 is output at the Q output.
- An F-STOP is triggered when there is an error in the safety data format in the associated instance DB. The following diagnostics event is then entered in the diagnostics buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

Response to F-STOP

Following an F-STOP, 16#00 is output at the QUALITY and STATUS outputs and QBAD.DATA = 1 is set. All other variables are frozen.

See also

Configuring fail-safe PA field devices (Page 61)

A.2.6.5 F_CH_DI: F-channel driver for digital inputs of F-I/O (except fail-safe DP standard slaves and fail-safe IO standard devices)

Function

The F-block is used for signal processing of a digital input value of an F-I/O (except fail-safe DP standard slaves and fail-safe IO standard devices). It supports channel-specific passivation and redundantly configured F-I/O.

The F-block cyclically reads the digital input value addressed at the VALUE input of an F-I/O from the associated F-module driver F_PS_12 that communicates with the F-I/O via a safety message frame according to the PROFIsafe bus profile. The F-module driver is automatically placed and interconnected with the CFC function "Generate module drivers".

If the digital input value is valid, it is made available at the Q output.

For redundantly configured F-I/O, the digital input value of the corresponding channel of the redundantly configured F-I/O is additionally read.

A quality code (QUALITY output) that can have the following states is generated for the result value at the Q output:

| State | Quality code (QUALITY output) |
|------------------------|-------------------------------|
| Valid value | 16#80 |
| Simulation | 16#60 |
| Fail-safe value | 16#48 |
| Invalid value (F-STOP) | 16#00 |

I/Os

| | Name | Data type | Explanation | Default |
|-----------------|----------|-----------|---|-------------------------------|
| Inputs: | ADR_CODE | DWORD | Coding for VALUE interconnection | to be automatically supplied* |
| | VALUE | BOOL | Address of the digital input channel | 0 |
| | SIM_I | F_BOOL | Simulation value | 0 |
| | SIM_ON | F_BOOL | 1 = Activate simulation value | 0 |
| | PASS_ON | F_BOOL | 1 = Activate passivation | 0 |
| | ACK_NEC | F_BOOL | 1 = User acknowledgement for reintegration after error required | 0 |
| | ACK_REI | F_BOOL | Acknowledgment for reintegration | 0 |
| Outputs: | PASS_OUT | F_BOOL | 1 = Passivation due to error | 0 |
| | QBAD | F_BOOL | 1 = Process value invalid | 0 |
| | QSIM | F_BOOL | 1 = Simulation active | 0 |

| | Name | Data type | Explanation | Default |
|--|---------|-----------|---|---------|
| | Q | F_BOOL | Process value | 0 |
| | QN | F_BOOL | Process value inverted | 1 |
| | Q_DATA | BOOL | DATA component of the process value (for visualization) | 0 |
| | QUALITY | BYTE | Value status (quality code) of the process value | 0 |
| | Q_MOD | BOOL | Value of F-I/O | 0 |
| | ACK_REQ | BOOL | Acknowledgment for reintegration required | 0 |
| | DISCF | BOOL | Discrepancy error on F-I/O | 0 |
| | DISCF_R | BOOL | Discrepancy error on redundant F-I/O | 0 |
| | QMODF | BOOL | 1 = F-I/O removed/faulty | 0 |
| | QMODF_R | BOOL | 1 = Redundant F-I/O removed/faulty | 0 |

*) The ADR_CODE input is automatically supplied when the S7 program is compiled and must not be changed. The ADR_CODE input is displayed as changed during the comparison of safety programs if changes were made to the address or the symbol of the signal at the VALUE input.

Note

Forcing the VALUE input

Forcing of the VALUE input is not possible because VALUE is not the value itself but rather a pointer for the address of the digital input channel.

Addressing

You must interconnect the symbol generated with *HW Config* in the symbol table for the digital input channel with the VALUE input.

Note

An inversion of the VALUE input in the *CFC Editor* is ineffective. Use the QN output instead.

Normal value

If the digital input value received from the F-I/O is valid, it is output at the Q output with quality code (QUALITY) 16#80.

Normal value for redundantly configured F-I/O

If both digital input values received from the redundantly configured F-I/O are valid, they are ORed and the result is output at the Q output with quality code (QUALITY) 16#80. If only one of the two received digital input values is valid, it is output at the Q output with quality code (QUALITY) 16#80.

Simulation

A simulation value can be output at the Q output instead of the normal value that is received from the F-I/O.

When input SIM_ON = 1, the value of the SIM_I input with quality code (QUALITY) 16#60 is output. Simulation has the highest priority. QBAD is always set to "0". If the F-block is in simulation state, QSIM = 1 is set.

When simulation is switched on, the digital input value received from the F-I/O is output at the Q_MOD output. If no communication is possible with the F-I/O or if there is still no user acknowledgement after an error, 0 is output. When simulation is switched off, Q_DATA is output.

Fail-safe value

The fail-safe value 0 is output at the Q output in the following cases:

- The digital input value is invalid due to a communication error (PROFIsafe).
- The digital input value is invalid due to a module or channel fault (e.g. wire break) or a fail-safe value is received from the module.
- For redundantly configured F-I/O: Both digital input values are invalid due to a communication error (PROFIsafe) or a module or channel fault (e.g. wire break).
- A passivation with PASS_ON = 1 is present.
- An F-startup is present.

The quality code (QUALITY) is set to 16#48 and QBAD = 1 is set.

If the output of the fail-safe value is not caused by a passivation, PASS_OUT = 1 is additionally set for passivation of other channels.

Reintegration after error elimination

After elimination of an error, the digital input value received from the F-I/O can be reintegrated automatically or only after user acknowledgment.

With parameter assignment ACK_NEC = 1, a user acknowledgement is required at the ACK_REI input after error elimination. With parameter assignment ACK_NEC = 0, reintegration occurs automatically.

For redundantly configured F-I/O, user acknowledgment is required even if the named errors occurred only on one F-I/O and therefore did not result in the output of a fail-safe value at the Q output.

Output ACK_REQ = 1 is used to signal that the error is eliminated and a user acknowledgment at the ACK_REI input is required for reintegration.

No user acknowledgement is required for a reintegration after PASS_ON = 1 of an F-startup after CPU STOP.

⚠ WARNING

Parameter assignment of input ACK_NEC = 0 is only allowed if an automatic reintegration is permissible under safety aspects for the process.

Communication errors (PROFIsafe) must always be acknowledged at the ACK_REI input independent of ACK_NEC. For this purpose, you must interconnect the ACK_REI input with a signal generated by an operator input. An interconnection with an automatically generated signal is not permissible.

⚠ WARNING**Startup protection for short-term power failure of the F-I/O**

After a power failure of the F-I/O that lasts for less time than the F-monitoring time (see section "Run times, F-Monitoring times, and response times (Page 460)") set for the F-I/O in *HW Config*, an automatic reintegration, as described for parameter assignment ACK_NEC = 0, may occur irrespective of your parameter assignment of the ACK_NEC input.

If automatic reintegration is not permitted for the process involved, you must program a startup protection for this situation by evaluating the QBAD or PASS_OUT outputs.

At a power failure of the F-I/O lasting longer than the F-monitoring time set for the F-I/O in *HW Config*, the F-system detects a communication error.

Discrepancy analysis for redundantly configured F-I/O

The F-block performs a discrepancy analysis for redundantly configured F-I/O when a discrepancy time $\neq 0$ was configured during redundancy configuration in *HW Config*.

If there is a discrepancy between the digital input channel addressed at the VALUE input and its redundant channel that lasts longer than the discrepancy time, a discrepancy error is detected. The F-block sets the DISCF output if the digital input channel addressed at the VALUE input supplies the 0 signal, or the DISCF_R output if the redundant channel supplies the 0 signal. DISCF/DISCF_R is reset as soon as a discrepancy is no longer present.

For example, the discrepancy analysis allows defective sensors to be detected because it is assumed that faulty fail-safe sensors supply the 0 signal. This can significantly increase the availability of the plant. Discrepancy errors have no effect on the Q, QBAD and PASS_OUT outputs. The non-fail-safe outputs DISCF/DISCF_R can be read out for service purposes via an OS or evaluated in the standard user program.

Startup behavior

After an F-startup, communication must first be established between the F-module driver and the F-I/O. During this time, the substitute value 0 with quality code (QUALITY output) 16#48 is output and outputs QBAD = 1 and PASS_OUT = 1 are additionally set. For redundantly configured F-I/O, the fail-safe value 0 is output until communication with one of the redundant F-I/O is established.

Error handling

An F-STOP is triggered when there is an error in the safety data format in the associated instance DB. The following diagnostics event is then entered in the diagnostics buffer of the F-CPU:

- "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

Response to F-STOP

Following an F-STOP, 16#00 is output at the QUALITY output and QBAD.DATA = 1 is set. All other variables are frozen.

A.2.6.6 F_CH_DO: F-channel driver for digital outputs of F-I/O (except fail-safe DP standard slaves and fail-safe IO standard devices)

Function

The F-block is used for signal processing of a digital output value of an F-I/O (except fail-safe DP standard slaves and fail-safe IO standard devices). It supports channel-specific passivation and redundantly configured F-I/O.

The F-block cyclically writes the digital output value for the output of an F-I/O addressed at the VALUE output to the associated F-module driver F_PS_12 that communicates with the F-I/O via a safety message frame according to the PROFIsafe bus profile. The F-module driver is automatically placed and interconnected with the CFC function "Generate module drivers".

For redundantly configured F-I/O, the digital output value is additionally written to the F-module driver of the redundantly configured F-I/O.

A quality code that can have the following states is generated for the digital output value that is written to the F-I/O:

| State | Quality code (QUALITY output) |
|------------------------|-------------------------------|
| Valid value | 16#80 |
| Simulation | 16#60 |
| Fail-safe value | 16#48 |
| Invalid value (F-STOP) | 16#00 |

I/Os

| | Name | Data type | Explanation | Default |
|-----------------|----------|-----------|---|-------------------------------|
| Inputs: | ADR_CODE | DWORD | Coding for VALUE interconnection | to be automatically supplied* |
| | I | F_BOOL | Process value | 0 |
| | SIM_I | F_BOOL | Simulation value | 0 |
| | SIM_MOD | F_BOOL | 1 = Simulation has priority | 0 |
| | SIM_ON | F_BOOL | 1 = Activate simulation value | 0 |
| | PASS_ON | F_BOOL | 1 = Activate passivation | 0 |
| | ACK_NEC | F_BOOL | 1 = User acknowledgement for reintegration after error required | 0 |
| | ACK_REI | F_BOOL | Acknowledgment for reintegration | 0 |
| Outputs: | PASS_OUT | F_BOOL | 1 = Passivation due to error | 0 |
| | QBAD | F_BOOL | 1 = Process value invalid | 0 |
| | QSIM | F_BOOL | 1 = Simulation active | 0 |
| | VALUE | BOOL | Address of the digital output channel | 0 |
| | QUALITY | BYTE | Value status (quality code) of the process value | 0 |
| | ACK_REQ | BOOL | Acknowledgment for reintegration required | 0 |
| | QMODF | BOOL | 1 = F-I/O removed/faulty | 0 |
| | QMODF_R | BOOL | 1 = Redundant F-I/O removed/faulty | 0 |

*) The ADR_CODE input is automatically supplied when the S7 program is compiled and must not be changed. The ADR_CODE input is displayed as changed during the comparison of safety programs if changes were made to the address or the symbol of the signal at the VALUE output.

Addressing

You must interconnect the symbol generated with *HW Config* in the symbol table for the digital output channel with the VALUE output.

Normal value

The process value active at the I input is written to the F-I/O. The quality code (QUALITY) is set to 16#80.

Normal value for redundantly configured F-I/O

For redundantly configured F-I/O, the process value active at the I input is written to both F-I/O, if both F-I/O have no communication error (PROFIsafe), no module or channel fault (e.g. wire break) and no F-startup. If one F-I/O has a communication error (PROFIsafe), module or channel fault (e.g. wire break) or an F-startup, the fail-safe value 0 is written to this F-I/O. The quality code (QUALITY) 16#80 is output.

Simulation

A simulation value can also be written to the F-I/O instead of the process value active at the I input.

When input SIM_ON = 1 and input SIM_MOD = 0, the value of the SIM_I input is written to the F-I/O and output at the VALUE output, if no communication error (PROFIsafe), no module or channel fault (e.g. wire break) and no F-stop is present.

When input SIM_ON = 1 and input SIM_MOD = 1, the value of the SIM_I input is output at the VALUE output even when a communication error (PROFIsafe), module or channel fault (e.g. wire break) or F-startup is present, in order to simulate an "error-free" operation without the presence of a real F-I/O.

In both cases, the quality code (QUALITY) is set to 16#60 and QSIM = 1 is set.

Fail-safe value

The fail-safe value 0 is written to the F-I/O when any of the following occurs:

- A communication error (PROFIsafe)
- A module or channel fault (e.g. wire break)
- An F-startup
- For redundantly configured F-I/O: a communication error (PROFIsafe), a module or channel fault (e.g. wire break) or an F-startup on both F-I/O
- Passivation with PASS_ON = 1

The quality code (QUALITY) is set to 16#48 and QBAD = 1 is set.

If the output of the fail-safe value is not caused by a passivation, PASS_OUT = 1 is additionally set for passivation of other channels.

Reintegration after error elimination

After elimination of an error, the F-I/O can be reintegrated automatically or only after user acknowledgment.

With parameter assignment ACK_NEC = 1, a user acknowledgement is required at the ACK_REI input after error elimination. With parameter assignment ACK_NEC = 0, reintegration occurs automatically.

For redundantly configured F-I/O, a user acknowledgment is also required under the following conditions:

- If the named errors occurred only on one F-I/O and therefore did not result in the output of a fail-safe value to the process.
- If communication errors (PROFIsafe) occurred on only one F-I/O and therefore did not result in the output of a fail-safe value to the process.

The acknowledgment of this error of the output module is made at the F_CH_DO F-channel driver for the whole module. After acknowledgment at an F-channel driver, all channels of the affected module are activated. To fully activate redundancy again for all channels, however, the error must also be acknowledged at the remaining "F_CH_DO" F-channel drivers.

The PROFISAFE output signals the presence of a communication error (PROFIsafe) at the module driver F_PS_12. This output can be used to enable automatic acknowledgment of all channels for this error. This requires that a corresponding acknowledgment is allowed for the process.

Output ACK_REQ = 1 is used to signal that the error is eliminated and a user acknowledgment at the ACK_REI input is required for reintegration.

No user acknowledgement is required for a reintegration after PASS_ON = 1 or an F-startup following a CPU STOP.

WARNING

Parameter assignment of input ACK_NEC = 0 is only allowed if an automatic reintegration is permissible under safety aspects for the process.

Communication errors (PROFIsafe) must always be acknowledged at the ACK_REI input independent of ACK_NEC. For this purpose, you must interconnect the ACK_REI input with a signal generated by an operator input. An interconnection with an automatically generated signal is not permissible.

WARNING

Startup protection for short-term power failure of the F-I/O

After a power failure of the F-I/O that lasts for less time than the F-monitoring time (see section "Run times, F-Monitoring times, and response times (Page 460)") set for the F-I/O in *HW Config*, an automatic reintegration, as described for parameter assignment ACK_NEC = 0, may occur irrespective of your parameter assignment of the ACK_NEC input.

If automatic reintegration is not permitted for the process involved, you must program a startup protection for this situation by evaluating the QBAD or PASS_OUT outputs.

At a power failure of the F-I/O lasting longer than the F-monitoring time set for the F-I/O in *HW Config*, the F-system detects a communication error.

Startup behavior

After an F-startup, communication must first be established between the F-module driver and the F-I/O. During this time, the fail-safe value 0 is written to the F-I/O. The quality code (QUALITY) is set to 16#48 and outputs QBAD = 1 and PASS_OUT = 1 are set. For redundantly configured F-I/O, the quality code (QUALITY) is set to 16#80 and outputs QBAD = 0 and PASS_OUT = 0 are set as soon as the communication with an F-I/O is established.

Error handling

An F-STOP is triggered when there is an error in the safety data format in the associated instance DB. The following diagnostics event is then entered in the diagnostics buffer of the F-CPU:

- "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

Response to F-STOP

Following an F-STOP, 16#00 is output at the QUALITY output and QBAD.DATA = 1 is set. All other variables are frozen.

A.2.6.7 F_CH_AI: F-channel driver for analog inputs of F-I/O (except fail-safe DP standard slaves and fail-safe IO standard devices)

Function

The F-block is used for signal processing of an analog input value of an F-I/O (except fail-safe DP standard slaves and fail-safe IO standard devices). It supports channel-specific passivation and redundantly configured I/O.

The F-block cyclically reads the analog input value (raw value) addressed at the VALUE input of an F-I/O from the associated F-module driver F_PS_12 that communicates with the F-I/O via a safety message frame according to the PROFIsafe bus profile. The F-module driver is automatically placed and interconnected with the CFC function "Generate module drivers".

If the analog input value is valid, it is adjusted to its physical quantity and made available at the V output.

For redundantly configured F-I/O, the analog input value of the corresponding channel of the redundantly configured F-I/O is additionally read.

A quality code (QUALITY output) that can have the following states is generated for the result value at the V output:

| State | Quality code (QUALITY output) |
|------------------------|-------------------------------|
| Valid value | 16#80 |
| Simulation | 16#60 |
| Fail-safe value | 16#48 |
| Last valid value | 16#44 |
| Invalid value (F-STOP) | 16#00 |

I/Os

| | Name | Data type | Explanation | Default |
|-----------------|------------|--|---|-------------------------------|
| Inputs: | ADR_CODE | DWORD | Coding for VALUE interconnection | to be automatically supplied* |
| | MODE | F_WORD | Measuring range coding | to be automatically supplied* |
| | VALUE | WORD | Address of the analog input channel | 0 |
| | VHRANGE | F_REAL | High limit of the process value | 0.0 |
| | VLRANGE | F_REAL | Low limit of the process value | 0.0 |
| | CH_F_ON | F_BOOL | 1 = Activate limit monitoring | 0 |
| | CH_F_HL | F_REAL | Overrange limit of the input value (mA) | 0.0 |
| | CH_F_LL | F_REAL | Underrange limit of the input value (mA) | 0.0 |
| | SIM_V | F_REAL | Simulation value | 0.0 |
| | SIM_ON | F_BOOL | 1 = Activate simulation value | 0 |
| | SUBS_V | F_REAL | Fail-safe value | 0.0 |
| | SUBS_ON | F_BOOL | 1 = Enable fail-safe value injection | 0 |
| | PASS_ON | F_BOOL | 1 = Activate passivation | 0 |
| | ACK_NEC | F_BOOL | 1 = User acknowledgement for reintegration after error required | 0 |
| | ACK_REI | F_BOOL | Acknowledgment for reintegration | 0 |
| | IPAR_EN ** | F_BOOL | 1 = Enable assignment of I-parameters | 0 |
| | IPAR_EN ** | F_BOOL | 1 = Enable assignment of I-parameters (redundant module) | 0 |
| | DISC_ON | BOOL | 1= Discrepancy analysis is performed | 0 |
| | DISC_TIME | DINT | Time in ms after which the discrepancy is displayed | 0 |
| DELTA | REAL | Maximum tolerable difference between the two modules | 0.0 | |
| Outputs: | PASS_OUT | F_BOOL | 1 = Passivation due to error | 0 |
| | QCHF_HL | F_BOOL | 1 = Input value in overrange | 0 |
| | QCHF_LL | F_BOOL | 1 = Input value in underrange | 0 |
| | QBAD | F_BOOL | 1 = Process value invalid | 0 |
| | QSIM | F_BOOL | 1 = Simulation active | 0 |
| | QSUBS | F_BOOL | 1 = Fail-safe value injection active | 0 |
| | OVHRANGE | F_REAL | High limit of the process value (copy) | 0.0 |
| | OVLRange | F_REAL | Low limit of the process value (copy) | 0.0 |
| | V | F_REAL | Process value | 0.0 |
| | V_DATA | REAL | DATA component of the process value (for visualization) | 0.0 |
| | QUALITY | BYTE | Value status (quality code) of the process value | 0 |
| | V_MOD | REAL | Value of F-I/O | 0.0 |
| | ACK_REQ | BOOL | Acknowledgment for reintegration required | 0 |

| Name | Data type | Explanation | Default |
|-------------|-----------|--|---------|
| IPAR_OK ** | F_BOOL | 1= New I-parameter values were assigned | |
| IPAR_OKR ** | F_BOOL | 1= New I-parameter values were assigned (redundant module) | |
| QMODF | BOOL | 1 = F-I/O removed/faulty | 0 |
| QMODF_R | BOOL | 1 = Redundant F-I/O removed/faulty | 0 |
| AL_STATE | STRUCT | Alarm status | |
| RAW_VALUE | WORD | Raw value | 0 |
| OVHRANGE | REAL | Copy of OVHRANGE | 0.0 |
| OVLRange | REAL | Copy of OVLRange | 0.0 |
| PASS_ON | BOOL | Copy of PASS_ON | 0 |
| PASS_OUT | BOOL | Copy of PASS_OUT | 0 |
| QCHF_HL | BOOL | Copy of QCHF_HL | 0 |
| QCHF_LL | BOOL | Copy of QCHF_LL | 0 |
| QBAD | BOOL | Copy of QBAD | 0 |
| QSIM | BOOL | Copy of QSIM | 0 |
| QSUBS | BOOL | Copy of QSUBS | 0 |
| ACK_REQ | BOOL | Copy of ACK_REQ | 0 |
| V_DATA | REAL | Copy of V_DATA | 0.0 |
| QUALITY | BYTE | Copy of QUALITY | 0 |
| V_MOD | REAL | Copy of V_MOD | 0.0 |
| DISCF | BOOL | Discrepancy error on F-I/O | 0 |

*) The ADR_CODE and MODE inputs are automatically supplied when the S7 program is compiled and must not be changed. The ADR_CODE input is displayed as changed during the comparison of safety programs if changes were made to the address or the symbol of the signal at the VALUE input. The MODE input is displayed as changed if changes in the configuration of the F-I/O have occurred.

**) These inputs/outputs are not visible. If you use this F-channel driver with an F-module with HART function, you may make these inputs/outputs visible and use them.

Note

Forcing the VALUE input

Forcing of the VALUE input is not possible because VALUE is not the value itself but rather a pointer for the address of the digital input channel.

Addressing

You must interconnect the symbol generated with *HW Config* in the symbol table for the analog input channel with the VALUE input.

Raw value check

Depending on the measuring mode and range, there is a nominal range of the F-I/O with analog inputs in which the analog signal is converted into a digitized raw value. There is also an overrange and underrange in which the analog signal can still be converted. Outside these limits, an overflow or underflow occurs. The F-channel driver indicates whether the raw value is within the nominal range of the F-I/O with analog inputs.

- When the nominal range is fallen below, output parameter QCHF_LL = 1 is set.
- When the nominal range is exceeded, output parameter QCHF_HL = 1 is set.

At an overflow or underflow, output QBAD = 1 is additionally set and – depending on the parameter assignment for the SUBS_ON input – the fail-safe value SUBS_V or the last valid value is output.

In the event of channel faults (e.g. wire break), 16#7FFF (overflow) or 16#8000 (underflow) is output by the F-I/O with analog inputs. Accordingly, the F-channel driver detects an overflow or underflow and sets outputs QCHF_HL or QCHF_LL = 1 and QBAD = 1.

(NAMUR) limit check for measuring range 4 to 20 mA

In the NAMUR guidelines for analog signal processing, limits are defined for Life Zero (4 to 20 mA) analog signals for which a channel fault is present:

$3.6 \text{ mA} < \text{Analog signal} < 21 \text{ mA}$.

The above NAMUR limits are set as fixed limits for the limit check by default. If other channel fault limits are to be set, input CH_F_ON = 1 must be set and the CH_F_HL and CH_F_LL inputs must be set with corresponding new limits in mA:

$\text{CH_F_LL} < \text{Analog signal} < \text{CH_F_HL}$

When the active channel fault limits are exceeded or fallen below, output QBAD = 1 is additionally set and – depending on the parameter assignment for the SUBS_ON input – the fail-safe value SUBS_V or the last valid value is output.

Note

The selectable limits must lie below the high limit of the overrange and above the low limit of the underrange of the F-I/O with analog inputs. Values outside the NAMUR range are thus also possible, if the F-I/O with analog inputs does not automatically limit the measured values to these.

Normal value

If the raw value received from the F-I/O is valid, it is adjusted to its physical quantity based on the VLRANGE and VHRANGE inputs and the measuring range coding and output at the V output with quality code (QUALITY) = 16#80.

In order for the settings of VLRANGE and VHRANGE to be interconnected with other block parameters, these settings are written to the OVLRange and OVHRANGE outputs.

The calculation algorithm assumes that the input signal is linear.

When VLRANGE = 0.0 and VHRANGE = 100.0, a percentage is output.

If VHRANGE = VLRANGE is set, the input signal of the F-I/O with analog inputs (e.g. mA value) is output according to the measuring range coding.

A parameter assignment of VHRANGE < VLRANGE is not permitted and results in invalid outputs.

Measuring range coding of the F-I/O with analog inputs

The measuring range is coded in *HW Config* by configuring the "Measuring range" parameter and, if necessary, the "Measuring method" parameter. The measuring range coding is automatically transferred to the MODE parameter of the F-channel driver. The F-channel driver supports the following measuring range codes:

| Measuring method | Measuring range | MODE (decimal/hex.) |
|------------------------------|-----------------|---------------------|
| 4-wire transducer | 0 to 20 mA | 514 / 16#0202 |
| or Measuring mode irrelevant | 4 to 20 mA | 515 / 16#0203 |
| 2-wire transducer | 4 to 20 mA | 771 / 16#0303 |

Normal value for redundantly configured F-I/O

For redundantly configured F-I/O, the raw value of the F-I/O that first supplies a valid value after an F-startup or initial run is output with quality code (QUALITY) = 16#80 at the V output after adjustment to its physical quantity. A changeover to the analog input value of the redundantly configured F-I/O then occurs if the currently output analog input value is invalid.

Simulation

A simulation value can be output at the V output instead of the normal value that is received from the F-I/O.

When input SIM_ON = 1, the value of the SIM_V input with quality code (QUALITY) 16#60 is output. Simulation has the highest priority. QSIM = 1 is set if the block is in simulation state.

Note

Ensure for the simulation that no invalid floating-point number (NaN) is present when there is an interconnected SIM_V input. This can be achieved, for example, by using the F-block F_LIM_R.

When VLRANGE = 0.0 and VHRANGE = 100.0, the value at the SIM_V input must be a percentage value.

So that the states of the QCHF_LL and QCHF_HL outputs can also be simulated, the simulation value is converted to a raw value based on the VHRANGE und VLRANGE inputs and the measuring range coding and checked like a raw value received from the F-I/O.

At an overflow or underflow of the active channel fault limits or when these are exceeded or fallen below (for measuring range 4-20 mA), the simulation value SIM_V is not output. Rather, the fail-safe value SUBS_V or the last valid value is output at the V output with quality code (QUALITY) 16#60, depending on the parameter assignment for the SUBS_ON input. QBAD = 1 is set.

When simulation is switched on, the analog input value received from the F-I/O is output as the process value at the V_MOD output. If no communication is possible with the F-I/O or if there is still no user acknowledgement after an error, 0.0 is output.

When simulation is switched off, V_DATA is output.

Fail-safe value

When input SUBS_ON = 1, the fail-safe value SUBS_V is output at the V output in the following cases:

- The analog input value is invalid due to a communication error (PROFIsafe).
- The analog input value is invalid due to a module or channel fault (e.g. wire break) or a fail-safe value is received from the module.
- The analog input value is invalid due to overflow or underflow.
- The analog input value is invalid because the active channel fault limits (for measuring range 4-20 mA) were exceeded or fallen below.
- For redundantly configured F-I/O: Both analog input values are invalid because of a communication error (PROFIsafe) or a module or channel fault (e.g. wire break) or because there is an overflow/underflow of the active channel limits or these were exceeded or fallen below (for measuring range 4-20 mA).
- A passivation with PASS_ON = 1 is present.
- An F-startup is present.

The quality code (QUALITY) is set to 16#48 and QSUBS = 1 and QBAD = 1 are set.

If the output of the fail-safe value is not caused by a passivation, PASS_OUT = 1 is additionally set for passivation of other channels.

Keep last value

When input SUBS_ON = 0, the last valid value of V is output at the V output in the following cases:

- The analog input value is invalid due to a communication error (PROFIsafe).
- The analog input value is invalid due to a module or channel fault (e.g. wire break) or a fail-safe value is received from the module.
- The analog input value is invalid due to overflow or underflow.
- The analog input value is invalid because the active channel fault limits (for measuring range 4-20 mA) were exceeded or fallen below.
- For redundantly configured F-I/O: Both analog input values are invalid because of a communication error (PROFIsafe) or a module or channel fault (e.g. wire break) or because there is an overflow/underflow of the active channel limits or these were exceeded or fallen below (for measuring range 4-20 mA).
- A passivation with PASS_ON = 1 is present.

The quality code (QUALITY) is set to 16#44 and QSUBS = 0 and QBAD = 1 are set.

If the output of the last valid value is not caused by a passivation, then output PASS_OUT = 1 is additionally set in order to passivate other channels.

Reintegration after error elimination


After elimination of an error, the analog input value received from the F-I/O can be reintegrated automatically or only after user acknowledgment.


With parameter assignment ACK_NEC = 1, a user acknowledgement is required at the ACK_REI input after error elimination. With parameter assignment ACK_NEC = 0, reintegration occurs automatically.

For redundantly configured F-I/O, user acknowledgment is required even if the named errors occurred only on one F-I/O and therefore did not result in the output of a fail-safe value at the V output.

Output ACK_REQ = 1 is used to signal that the error is eliminated and a user acknowledgment at the ACK_REI input is required for reintegration.

No user acknowledgement is required for a reintegration after PASS_ON = 1 or an F-startup following a CPU STOP.

| |
|---|
|  WARNING |
| Parameter assignment of input ACK_NEC = 0 is only allowed if an automatic reintegration is permissible under safety aspects for the process. |
| Communication errors (PROFIsafe) must always be acknowledged at the ACK_REI input independent of ACK_NEC. For this purpose, you must interconnect the ACK_REI input with a signal generated by an operator input. An interconnection with an automatically generated signal is not permissible. |

| |
|--|
|  WARNING |
| Startup protection for short-term power failure of the F-I/O |
| After a power failure of the F-I/O that lasts for less time than the F-monitoring time (see section "Run times, F-Monitoring times, and response times (Page 460)") set for the F-I/O in <i>HW Config</i> , an automatic reintegration, as described for parameter assignment ACK_NEC = 0, may occur irrespective of your parameter assignment of the ACK_NEC input. |
| If automatic reintegration is not permitted for the process involved, you must program a startup protection for this situation by evaluating the QBAD or PASS_OUT outputs. |
| At a power failure of the F-I/O lasting longer than the F-monitoring time set for the F-I/O in <i>HW Config</i> , the F-system detects a communication error. |

Configurable alarm limits

The raw value and inputs/outputs of the F-channel driver are additionally bundled in a structure and made available as non-fail-safe information at the AL_STATE output. This allows you to evaluate configurable alarm limits in the standard user program. The mapping to a structure allows the information to be exchanged between an F-channel driver and a standard block via a single interconnection.

Parameter reassignment of an F-I/O

For parameter reassignment of an F-I/O, the IPAR_EN input and the IPAR_OK output are available. The IPAR_EN input and the IPAR_OK output are not visible. When an F-module with HART function is used, make the input and output visible.

The IPAR_EN input corresponds to the iPar_EN_C variable and the IPAR_OK output corresponds to the iPar_OK_S variable in the PROFIsafe bus profile as of PROFIsafe Specification V1.30. To find out when you must set/reset the IPAR_EN input following parameter reassignment of an F-I/O and how you can evaluate the IPAR_OK output, refer to the PROFIsafe Specification V1.30 or higher and the documentation for the F-I/O.

If more than one F-channel driver is placed for an F-I/O, iPar_EN_C is formed from an OR logic operation of all IPAR_EN of the F-channel drivers belonging to the F-I/O.

If passivation is to be carried out when IPAR_EN = 1, you must additionally set variable PASS_ON = 1.

For the redundantly configured F-I/O, the IPAR_ENR/IPAR_OKR signals are used.

The IPAR_EN and IPAR_ENR inputs and the IPAR_OK and IPAR_OKR outputs are not visible. When an F-module with HART function is used, make the input and output visible.

For an F-module with HART function, the IPAR_EN/IPAR_OK inputs/outputs are used for deactivation of the HART protocol. For a detailed description of how the signals are processed in the safety program, refer to the manual of the F-module, e.g. SM 336; F-AI 6 x 0/4 ... 20 mA HART, on the Internet (<http://support.automation.siemens.com/WW/view/en/19026151>).

Discrepancy analysis for redundantly configured F-I/O

The F-block performs a discrepancy analysis for redundantly configured F-I/O. For this purpose, the following must be configured in the CFC:

- DISC_ON must be set in order to activate the discrepancy analysis
- A value must be entered for DISC_TIME. The value specifies how long the discrepancy must last before it is indicated.
- For DELTA, a value must be set that specifies the maximum discrepancy. If the discrepancy > DELTA, the discrepancy error is detected.

If there is a discrepancy between the analog input channel addressed at the VALUE input and its redundant channel that lasts longer than the discrepancy time DISC_TIME, a discrepancy error is detected. The F-block sets the DISCF output if the analog input channel addressed at the VALUE input deviates from the redundant channel by a value > DELTA. DISCF is reset as soon as a discrepancy is no longer present.

Discrepancy errors have no effect on the V, QBAD and PASS_OUT outputs. The non-fail-safe output DISCF can be read out for service purposes via an OS or evaluated in the standard user program.

Startup behavior

After an F-startup, communication must first be established between the F-module driver and the F-I/O. During this time, the fail-safe value SUBS_V with quality code (QUALITY output) 16#48 is output irrespective of the parameter assignment at the SUBS_ON input and outputs QSUBS = 1, QBAD = 1 and PASS_OUT = 1 are additionally set. For redundantly configured F-I/O, the fail-safe value SUBS_V is output until communication with one of the redundant F-I/O is established.

Error handling

- If measuring range coding at the MODE input is not supported, an invalid raw value is assumed.
- If one of the VHRANGE, VLRANGE, CH_F_HL, CH_F_LL, and SUBS_V inputs is an invalid floating-point number (NaN) or if invalid floating-point numbers (NaN) result from the calculation in the F-block, the fail-safe value SUBS_V or the last valid value is output at the V output, depending on the parameter assignment for the SUBS_ON input. The QBAD, QCHF_LL and QCHF_HL outputs are set to 1. Quality code (QUALITY) and QSUBS are formed appropriately for this.

For the SIM_V input, note the information under "Simulation".

For the case that invalid floating-point numbers (NaN) result from the calculation in the F-block, the following diagnostics event is entered in the diagnostics buffer of the CPU:

- "Safety program: Invalid REAL number in DB" (Event ID 16#75D9).
- An F-STOP is triggered when there is an error in the safety data format in the associated instance DB. The following diagnostics event is then entered in the diagnostics buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

Response to F-STOP

Following an F-STOP, 16#00 is output at the QUALITY output and QBAD.DATA = 1 is set. All other variables are frozen.

A.2.6.8 F_CH_II: F-Channel driver for inputs of data type INT of fail-safe DP standard slaves and fail-safe standard I/O devices

Function

The F-block is used for signal processing of an input value of data type INT of fail-safe DP standard slaves/fail-safe IO standard devices.

The F-block cyclically reads the input value addressed at the VALUE input of data type INT of a fail-safe DP standard slave/IO standard device from the associated F-module driver F_PS_12 that communicates with the fail-safe DP standard slave/IO standard device via a safety message frame according to the PROFIsafe bus profile. The F-module driver is automatically placed and interconnected with the CFC function "Generate module drivers".

If the input value is valid, it is made available at the V output as F_Real and at the V_INT output as Integer.

A quality code (QUALITY output) that can have the following states is generated for the result value at the V output:

| State | Quality code (QUALITY output) |
|------------------------|-------------------------------|
| Valid value | 16#80 |
| Simulation | 16#60 |
| Last valid value | 16#44 |
| Fail-safe value | 16#48 |
| Invalid value (F-STOP) | 16#00 |

I/Os

| | Name | Data type | Explanation | Default |
|-----------------|----------|-----------|---|-------------------------------|
| Inputs: | ADR_CODE | DWORD | Coding for VALUE interconnection | To be automatically supplied* |
| | VALUE | INT | Address of the input channel | 0 |
| | SIM_V | F_INT | Simulation value | 0 |
| | SIM_ON | F_BOOL | 1=Activate simulation value | 0 |
| | SUBS_V | F_INT | Fail-safe value | 0 |
| | SUBS_ON | F_BOOL | 1=Enable fail-safe value injection | 0 |
| | PASS_ON | F_BOOL | 1=Enable passivation | 0 |
| | ACK_NEC | F_BOOL | 1=User acknowledgement for reintegration after error required | 0 |
| | ACK_REI | F_BOOL | Acknowledgement for reintegration | 0 |
| | IPAR_EN | F_BOOL | 1=Enable assignment of I-parameters | 0 |
| Outputs: | PASS_OUT | F_BOOL | 1=Passivation due to error | 0 |
| | QBAD | F_BOOL | 1=Process value invalid | 0 |
| | QSIM | F_BOOL | 1=Simulation active | 0 |
| | QSUBS | F_BOOL | 1=Fail-safe value injection active | 0 |
| | V | F_REAL | Process value | 0.0 |
| | V_DATA | REAL | DATA component of the process value (for visualization) | 0.0 |

| | Name | Data type | Explanation | Default |
|--|---------|-----------|--|---------|
| | QUALITY | BYTE | Value status (quality code) of the process value | B#16#0 |
| | V_INT | F_INT | INT process value | 0 |
| | V_MOD | REAL | Value of F-I/O | 0.0 |
| | ACK_REQ | BOOL | Acknowledgement for reintegration required | 0 |
| | IPAR_OK | F_BOOL | 1= New I-parameter values were assigned | 0 |

*) The ADR_CODE input is automatically supplied when the S7 program is compiled and must not be changed. The ADR_CODE input is displayed as changed during the comparison of safety programs if changes were made to the address or the symbol of the signal at the VALUE input.

Note

Forcing the VALUE input

Forcing of the VALUE input is not possible because VALUE is not the value itself but rather a pointer for the address of the digital input channel.

Addressing

You must interconnect the symbol generated with *HW Config* in the symbol table for the input value of data type INT with the VALUE input.

Normal value

If the input value received from the fail-safe DP standard slave/I/O standard device is valid, it is output at the V and V_INT output with quality code (QUALITY) 16#80.

Simulation

A simulation value can be output at the V and V_INT output instead of the normal value that is received from the fail-safe DP standard slave/I/O standard device.

When input SIM_ON = 1, the value of the SIM_V input with quality code (QUALITY) 16#60 is output. Simulation has the highest priority. QBAD is always set to "0". QSIM = 1 is set if the block is in simulation state.

When simulation is switched on, the input value received from the fail-safe DP standard slave/I/O standard device is output at the V_MOD output. If no communication is possible with the fail-safe DP standard slave/I/O standard device or if there is still no user acknowledgement after an error, 0 is output.

When simulation is switched off, V_DATA is output.

Fail-safe value

When input SUBS_ON = 1, the fail-safe value SUBS_V is output at the V and V_INT output in the following cases:

- The input value is invalid due to a communication error (PROFIsafe).
- The input value is invalid due to a module fault or a fail-safe value is received from the module.
- A passivation with PASS_ON = 1 is present.
- An F-startup is present.
- FV_ACTIVATED is signaled by the module

The quality code (QUALITY) is set to 16#48 and QSUBS = 1 and QBAD = 1 are set.

If the output of the fail-safe value is not caused by a passivation, PASS_OUT = 1 is additionally set for passivation of other channels.

Keep last value

When input SUBS_ON = 0, the last valid value of V is output at the V and V_INT output:

- The input value is invalid due to a communication error (PROFIsafe) or a fail-safe value is received from the module.
- The input value is invalid due to a module fault or a fail-safe value is received from the module.
- A passivation with PASS_ON = 1 is present.
- FV_ACTIVATED is signaled by the module

The quality code (QUALITY) is set to 16#44 and QSUBS = 0 and QBAD = 1 are set.

If the output of the last valid value is not caused by a passivation, then output PASS_OUT = 1 is additionally set in order to passivate other channels.

Reintegration

After elimination of an error, the input value received from the fail-safe DP standard slave/IO standard device can be reintegrated automatically or only after user acknowledgment.

With parameter assignment ACK_NEC = 1, a user acknowledgement is required at the ACK_REI input after error elimination. With parameter assignment ACK_NEC = 0, reintegration occurs automatically.

Output ACK_REQ = 1 is used to signal that the error is eliminated and a user acknowledgment at the ACK_REI input is required for reintegration.

No user acknowledgement is required for reintegration after $PASS_ON = 1$. For reintegration after an F-startup following a CPU STOP, no user acknowledgment is required if the F-I/O starts up using the Slave State (20) "system start" according to the PROFIsafe Specification V1.30 and higher. Otherwise, a communication error (PROFIsafe) is detected.

 **WARNING**

Assignment of input $ACK_NEC = 0$ is only allowed if an automatic reintegration is permissible under safety aspects for the process.

Communication errors (PROFIsafe) must always be acknowledged at the ACK_REI input independent of ACK_NEC . For this purpose, you must interconnect the ACK_REI input with a signal generated by an operator input. An interconnection with an automatically generated signal is not permissible.

 **WARNING**

Startup protection for short-term power failure of the fail-safe DP standard slave/IO standard device

After a power failure of the fail-safe DP standard slave/IO standard device that lasts for less time than the F-monitoring time (see section "Run times, F-Monitoring times, and response times (Page 460)") set for the fail-safe DP standard slave/IO standard device in *HW Config*, an automatic reintegration, as described for parameter assignment $ACK_NEC = 0$, may occur irrespective of your parameter assignment of the ACK_NEC input.

If automatic reintegration is not permitted for the process involved, you must program a startup protection for this situation by evaluating the $QBAD$ or $PASS_OUT$ outputs.

At a power failure of the F-I/O lasting longer than the F-monitoring time set for the F-I/O in *HW Config*, the F-system detects a communication error.

Parameter reassignment of a fail-safe DP standard slave/IO standard device

For parameter reassignment of a fail-safe DP standard slave/IO standard device, the $IPAR_EN$ input and the $IPAR_OK$ output are available.

The $IPAR_EN$ input corresponds to the $iPar_EN_C$ variable and the $IPAR_OK$ output corresponds to the $iPar_OK_S$ variable in the PROFIsafe bus profile as of PROFIsafe Specification V1.30. To find out when you must set/reset the $IPAR_EN$ input following parameter reassignment of a fail-safe DP standard slave/IO standard device and how you can evaluate the $IPAR_OK$ output, refer to the PROFIsafe Specification V1.30 or higher and the documentation for the fail-safe standard DP slave/IO standard device.

If more than one F-channel driver is placed for a fail-safe DP standard slave/IO standard device, $iPar_EN_C$ is formed from an OR logic operation of all $IPAR_EN$ of the F-channel drivers belonging to the fail-safe DP standard slave/IO standard device.

If passivation is to be carried out when $IPAR_EN = 1$, you must additionally set variable $PASS_ON = 1$.

Startup behavior

After an F-startup, communication must first be established between the F-module driver and the fail-safe DP standard slave/IO standard device. During this time, the fail-safe value SUBS_V with quality code (QUALITY output) 16#48 is output irrespective of the parameter assignment at the SUBS_ON input and outputs QSUBS = 1, QBAD = 1 and PASS_OUT = 1 are additionally set.

Error handling

- An F-STOP is triggered when there is an error in the safety data format in the associated instance DB. The following diagnostics event is then entered in the diagnostics buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

Response to F-STOP

Following an F-STOP, 16#00 is output at the QUALITY output and QBAD.DATA = 1 is set. All other variables are frozen.

A.2.6.9 F_CH_IO: F-Channel driver for outputs of data type INT of fail-safe DP standard slaves and fail-safe standard I/O devices

Function

The F-block is used for signal processing of an output value of data type INT of fail-safe DP standard slaves/fail-safe IO standard devices.

The block cyclically writes the output value of data type INT for the output of a fail-safe DP standard slave/IO standard device addressed at the VALUE output to the associated F-module driver F_PS_12 that communicates with the fail-safe DP standard slave/IO standard device via a safety message frame according to the PROFIsafe bus profile. The F-module driver is automatically placed and interconnected with the CFC function "Generate module drivers".

A quality code that can have the following states is generated for the output value that is written to the fail-safe DP standard slave/IO standard device:

| State | Quality code (QUALITY output) |
|------------------------|-------------------------------|
| Valid value | 16#80 |
| Simulation | 16#60 |
| Fail-safe value | 16#48 |
| Invalid value (F-STOP) | 16#00 |

I/Os

| | Name | Data type | Explanation | Default |
|-----------------|----------|-------------------------------------|---|-------------------------------|
| Inputs: | ADR_CODE | DWORD | Coding for VALUE interconnection | To be automatically supplied* |
| | I | F_INT | Process value | 0 |
| | SIM_I | F_INT | Simulation value | 0 |
| | SIM_MOD | F_BOOL | 1=Simulation value takes precedence | 0 |
| | SIM_ON | F_BOOL | 1=Activate simulation value | 0 |
| | PASS_ON | F_BOOL | 1=Enable passivation | 0 |
| | ACK_NEC | F_BOOL | 1=User acknowledgement for reintegration after error required | 0 |
| | ACK_REI | F_BOOL | Acknowledgement for reintegration | 0 |
| IPAR_EN | F_BOOL | 1=Enable assignment of I-parameters | 0 | |
| | | | | |
| Outputs: | PASS_OUT | F_BOOL | 1=Passivation due to error | 0 |
| | QBAD | F_BOOL | 1=Process value invalid | 0 |
| | QSIM | F_BOOL | 1=Simulation active | 0 |
| | VALUE | INT | Address of the output channel | 0 |
| | QUALITY | BYTE | Value status (quality code) of the process value | B#16#0 |
| | ACK_REQ | BOOL | Acknowledgement for reintegration required | 0 |
| | IPAR_OK | F_BOOL | 1= New I-parameter values were assigned | 0 |

*) The ADR_CODE input is automatically supplied when the S7 program is compiled and must not be changed. The ADR_CODE input is displayed as changed during the comparison of safety programs if changes were made to the address or the symbol of the signal at the VALUE input.

Addressing

You must interconnect the symbol generated with *HW Config* in the symbol table for the input value of data type INT with the VALUE input.

Normal value

The process value active at the I input is written to the fail-safe DP standard slave/IO standard device. The quality code (QUALITY) is set to 16#80.

Fail-safe value

The fail-safe value 0 is written to the fail-safe DP standard slave/IO standard device when any of the following occurs:

- A communication error (PROFIsafe)
- A module or channel fault (e.g. wire break)
- An F-startup
- Passivation with PASS_ON = 1

The quality code (QUALITY) is set to 16#48 and QBAD = 1 is set.

If the output of the fail-safe value is not caused by a passivation, PASS_OUT = 1 is additionally set for passivation of other channels.

Note

For fail-safe DP standard slaves/IO standard devices, channel-specific passivation is not possible for outputs using PASS_ON. If you have placed more than one F-channel driver for outputs for a fail-safe DP standard slave/IO standard device, the fail-safe value 0 is written for all outputs of the fail-safe DP standard slave/IO standard device following a passivation with PASS_ON = 1 at one of the F-channel drivers. If you want to evaluate the QBAD and QUALITY outputs of the other F-channel drivers when PASS_ON = 1 at one of the F-channel drivers, you must control the PASS_ON inputs of all F-channel drivers synchronously.

Simulation

A simulation value can also be written to the fail-safe DP standard slave/IO standard device instead of the process value active at the I input.

When input SIM_ON = 1 and input SIM_MOD = 0, the value of the SIM_I input is written to the fail-safe DP standard slave/IO standard device and output at the VALUE output, if no communication error (PROFIsafe), no module or channel fault (e.g. wire break) and no F-startup is present.

When input SIM_ON = 1 and input SIM_MOD = 1, the value of the SIM_I input is output at the VALUE output even when a communication error (PROFIsafe), module or channel fault (e.g. wire break) or F-startup is present, in order to simulate an "error-free" operation without the presence of a real fail-safe DP standard slave/IO standard device.

In both cases, the quality code (QUALITY) is set to 16#60 and QSIM = 1 is set.

Note

If you have placed more than one F-channel driver for outputs for a fail-safe DP standard slave/IO standard device, a simulation value is not written if the PASS_ON input of another F-channel driver for outputs of the fail-safe DP standard slave/IO standard device is 1 and the SIM_ON input is 0.

Reintegration

After elimination of an error, the fail-safe DP standard slave/IO standard device can be reintegrated automatically or only after user acknowledgment.

With parameter assignment `ACK_NEC = 1`, a user acknowledgement is required at the `ACK_REI` input after error elimination. With parameter assignment `ACK_NEC = 0`, reintegration occurs automatically.

Output `ACK_REQ = 1` is used to signal that the error is eliminated and a user acknowledgment at the `ACK_REI` input is required for reintegration.

No user acknowledgement is required for reintegration after `PASS_ON = 1`. For reintegration after an F-startup following a CPU STOP, no user acknowledgment is required if the fail-safe DP standard slave/IO standard device starts up using the Slave State (20) "system start" according to the PROFIsafe Specification V1.30 and higher. Otherwise, a communication error (PROFIsafe) is detected.

Note

For fail-safe DP standard slaves/IO standard devices, channel-specific reintegration is not possible for outputs using `PASS_ON`. If you have placed more than one F-channel driver for outputs for a fail-safe DP standard slave/IO standard device, you must synchronously control the `ACK_REI` inputs of all F-channel drivers for outputs of the fail-safe DP standard slave/IO standard device.



WARNING

Assignment of input `ACK_NEC = 0` is only allowed if an automatic reintegration is permissible under safety aspects for the process.

Communication errors (PROFIsafe) must always be acknowledged at the `ACK_REI` input independent of `ACK_NEC`. For this purpose, you must interconnect the `ACK_REI` input with a signal generated by an operator input. An interconnection with an automatically generated signal is not permissible.



WARNING

Startup protection for short-term power failure of the fail-safe DP standard slave/IO standard device

After a power failure of the fail-safe DP standard slave/IO standard device that lasts for less time than the F-monitoring time (see section "Run times, F-Monitoring times, and response times (Page 460)") set for the fail-safe DP standard slave/IO standard device in *HW Config*, an automatic reintegration, as described for parameter assignment `ACK_NEC = 0`, may occur irrespective of your parameter assignment of the `ACK_NEC` input.

If automatic reintegration is not permitted for the process involved, you must program a startup protection for this situation by evaluating the `QBAD` or `PASS_OUT` outputs.

At a power failure of the F-I/O lasting longer than the F-monitoring time set for the F-I/O in *HW Config*, the F-system detects a communication error.

Parameter reassignment of a fail-safe DP standard slave/IO standard device

For parameter reassignment of a fail-safe DP standard slave/IO standard device, the IPAR_EN input and the IPAR_OK output are available.

The IPAR_EN input corresponds to the iPar_EN_C variable and the IPAR_OK output corresponds to the iPar_OK_S variable in the PROFIsafe bus profile as of PROFIsafe Specification V1.30. To find out when you must set/reset the IPAR_EN input following parameter reassignment of a fail-safe DP standard slave/IO standard device and how you can evaluate the IPAR_OK output, refer to the PROFIsafe Specification V1.30 or higher and the documentation for the fail-safe standard DP slave/IO standard device.

If more than one F-channel driver is placed for a fail-safe DP standard slave/IO standard device, iPar_EN_C is formed from an OR logic operation of all IPAR_EN of the F-channel drivers belonging to the fail-safe DP standard slave/IO standard device.

If passivation is to be carried out when IPAR_EN = 1, you must additionally set variable PASS_ON = 1.

Startup behavior

After an F-startup, communication must first be established between the F-module driver and the fail-safe DP standard slave/IO standard device. During this time, the fail-safe value 0 is written to the fail-safe DP standard slave/IO standard device. The quality code (QUALITY) is set to 16#48 and outputs QBAD = 1 and PASS_OUT = 1 are set.

Error handling

- An F-STOP is triggered when there is an error in the safety data format in the associated instance DB. The following diagnostics event is then entered in the diagnostics buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

Response to F-STOP

Following an F-STOP, 16#00 is output at the QUALITY output and QBAD.DATA = 1 is set. All other variables are frozen.

A.2.6.10 F_CH_DII: F-Channel driver for inputs of data type DINT of fail-safe DP standard slaves and fail-safe standard I/O devices

Function

The F-block is used for signal processing of an input value of data type DINT of fail-safe DP standard slaves/fail-safe IO standard devices.

The F-block cyclically reads the input value addressed at the VALUE input of data type DINT of a fail-safe DP standard slave/IO standard device from the associated F-module driver F_PS_12 that communicates with the fail-safe DP standard slave/IO standard device via a safety message frame according to the PROFIsafe bus profile. The F-module driver is automatically placed and interconnected with the CFC function "Generate module drivers".

If the input value is valid, it is made available at the V output as F_Real and at the V_DINT output of data type F_DINT.

Note

When values are converted from F_DINT to F_REAL, an inaccuracy of up to 127 arises for values > +16,777,215 or < -16,777,216. That is, the value in F_DINT format is rounded up or down for representation in F_REAL format, because 8 bits of the 32-bit REAL value are needed for representing the exponent.

A quality code (QUALITY output) that can have the following states is generated for the result value at the V output:

| State | Quality code (QUALITY output) |
|------------------------|-------------------------------|
| Valid value | 16#80 |
| Simulation | 16#60 |
| Last valid value | 16#44 |
| Fail-safe value | 16#48 |
| Invalid value (F-STOP) | 16#00 |

I/Os

| | Name | Data type | Explanation | Default |
|-----------------|----------|-----------|---|-------------------------------|
| Inputs: | ADR_CODE | DWORD | Coding for VALUE interconnection | To be automatically supplied* |
| | VALUE | DINT | Address of the input channel | 0 |
| | SIM_V | F_DINT | Simulation value | 0 |
| | SIM_ON | F_BOOL | 1=Activate simulation value | 0 |
| | SUBS_V | F_DINT | Fail-safe value | 0 |
| | SUBS_ON | F_BOOL | 1=Enable fail-safe value injection | 0 |
| | PASS_ON | F_BOOL | 1=Enable passivation | 0 |
| | ACK_NEC | F_BOOL | 1=User acknowledgement for reintegration after error required | 0 |
| | ACK_REI | F_BOOL | Acknowledgement for reintegration | 0 |
| | IPAR_EN | F_BOOL | 1=Enable assignment of I-parameters | 0 |
| Outputs: | PASS_OUT | F_BOOL | 1=Passivation due to error | 0 |
| | QBAD | F_BOOL | 1=Process value invalid | 0 |
| | QSIM | F_BOOL | 1=Simulation active | 0 |
| | QSUBS | F_BOOL | 1=Fail-safe value injection active | 0 |
| | V | F_REAL | Process value | 0.0 |

| | Name | Data type | Explanation | Default |
|--|---------|-----------|---|---------|
| | V_DATA | REAL | DATA component of the process value (for visualization) | 0.0 |
| | QUALITY | BYTE | Value status (quality code) of the process value | B#16#0 |
| | V_DINT | F_DINT | DINT process value | 0 |
| | V_MOD | REAL | Value of F-I/O | 0.0 |
| | ACK_REQ | BOOL | Acknowledgement for reintegration required | 0 |
| | IPAR_OK | F_BOOL | 1= New I-parameter values were assigned | 0 |

*) The ADR_CODE input is automatically supplied when the S7 program is compiled and must not be changed. The ADR_CODE input is displayed as changed during the comparison of safety programs if changes were made to the address or the symbol of the signal at the VALUE input.

Note

Forcing the VALUE input

Forcing of the VALUE input is not possible because VALUE is not the value itself but rather a pointer for the address of the digital input channel.

Addressing

You must interconnect the symbol generated with *HW Config* in the symbol table for the input value of data type DINT with the VALUE input.

Normal value

If the input value received from the fail-safe DP standard slave/IO standard device is valid, it is output at the V and V_INT output with quality code (QUALITY) 16#80.

Simulation

A simulation value can be output at the V and V_DINT output instead of the normal value that is received from the fail-safe DP standard slave/IO standard device.

When input SIM_ON = 1, the value of the SIM_V input with quality code (QUALITY) 16#60 is output. Simulation has the highest priority. QBAD is always set to "0". QSIM = 1 is set if the block is in simulation state.

When simulation is switched on, the input value received from the fail-safe DP standard slave/IO standard device is output at the V_MOD output. If no communication is possible with the fail-safe DP standard slave/IO standard device or if there is still no user acknowledgement after an error, 0 is output.

When simulation is switched off, V_DATA is output.

Fail-safe value

When input SUBS_ON = 1, the fail-safe value SUBS_V is output at the V and V_DINT output in the following cases:

- The input value is invalid due to a communication error (PROFIsafe).
- The input value is invalid due to a module fault or a fail-safe value is received from the module.
- A passivation with PASS_ON = 1 is present.
- An F-startup is present.
- FV_ACTIVATED is signaled by the module

The quality code (QUALITY) is set to 16#48 and QSUBS = 1 and QBAD = 1 are set.

If the output of the fail-safe value is not caused by a passivation, PASS_OUT = 1 is additionally set for passivation of other channels.

Keep last value

When input SUBS_ON = 0, the last valid value of V is output at the V and V_DINT output in the following cases:

- The input value is invalid due to a communication error (PROFIsafe).
- The input value is invalid due to a module fault or a fail-safe value is received from the module.
- A passivation with PASS_ON = 1 is present.
- FV_ACTIVATED is signaled by the module

The quality code (QUALITY) is set to 16#44 and QSUBS = 0 and QBAD = 1 are set.

If the output of the last valid value is not caused by a passivation, then output PASS_OUT = 1 is additionally set in order to passivate other channels.

Reintegration

After elimination of an error, the input value received from the fail-safe DP standard slave/IO standard device can be reintegrated automatically or only after user acknowledgment.

With parameter assignment ACK_NEC = 1, a user acknowledgement is required at the ACK_REI input after error elimination. With parameter assignment ACK_NEC = 0, reintegration occurs automatically.

Output ACK_REQ = 1 is used to signal that the error is eliminated and a user acknowledgment at the ACK_REI input is required for reintegration.

No user acknowledgement is required for reintegration after PASS_ON = 1. For reintegration after an F-startup following a CPU STOP, no user acknowledgment is required if the F-I/O starts up using the Slave State (20) "system start" according to the PROFIsafe Specification V1.30 and higher. Otherwise, a communication error (PROFIsafe) is detected.

 **WARNING**

Assignment of input ACK_NEC = 0 is only allowed if an automatic reintegration is permissible under safety aspects for the process.

Communication errors (PROFIsafe) must always be acknowledged at the ACK_REI input independent of ACK_NEC. For this purpose, you must interconnect the ACK_REI input with a signal generated by an operator input. An interconnection with an automatically generated signal is not permissible.

 **WARNING**

Startup protection for short-term power failure of the fail-safe DP standard slave/IO standard device

After a power failure of the fail-safe DP standard slave that lasts for less time than the F-monitoring time (see section "Run times, F-Monitoring times, and response times (Page 460)") set for the fail-safe DP standard slave/IO standard device in *HW Config*, an automatic reintegration, as described for parameter assignment ACK_NEC = 0, may occur irrespective of your parameter assignment of the ACK_NEC input.

If automatic reintegration is not permitted for the process involved, you must program a startup protection for this situation by evaluating the QBAD or PASS_OUT outputs.

At a power failure of the F-I/O lasting longer than the F-monitoring time set for the F-I/O in *HW Config*, the F-system detects a communication error.

Parameter reassignment of a fail-safe DP standard slave/IO standard device

For parameter reassignment of a fail-safe DP standard slave/IO standard device, the IPAR_EN input and the IPAR_OK output are available.

The IPAR_EN input corresponds to the iPar_EN_C variable and the IPAR_OK output corresponds to the iPar_OK_S variable in the PROFIsafe bus profile as of PROFIsafe Specification V1.30. To find out when you must set/reset the IPAR_EN input following parameter reassignment of a fail-safe DP standard slave/IO standard device and how you can evaluate the IPAR_OK output, refer to the PROFIsafe Specification V1.30 or higher and the documentation for the fail-safe standard DP slave/IO standard device.

If more than one F-channel driver is placed for a fail-safe DP standard slave/IO standard device, iPar_EN_C is formed from an OR logic operation of all IPAR_EN of the F-channel drivers belonging to the fail-safe DP standard slave/IO standard device.

If passivation is to be carried out when IPAR_EN = 1, you must additionally set variable PASS_ON = 1.

Startup behavior

After an F-startup, communication must first be established between the F-module driver and the fail-safe DP standard slave/IO standard device. During this time, the fail-safe value SUBS_V with quality code (QUALITY output) 16#48 is output irrespective of the parameter assignment at the SUBS_ON input and outputs QSUBS = 1, QBAD = 1 and PASS_OUT = 1 are additionally set.

Error handling

- An F-STOP is triggered when there is an error in the safety data format in the associated instance DB. The following diagnostics event is then entered in the diagnostics buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

Response to F-STOP

Following an F-STOP, 16#00 is output at the QUALITY output and QBAD.DATA = 1 is set. All other variables are frozen.

A.2.6.11 F_CH_DIO: F-Channel driver for outputs of data type DINT of fail-safe DP standard slaves and fail-safe standard I/O devices

Function

The F-block is used for signal processing of an output value of data type DINT of fail-safe DP standard slaves/fail-safe IO standard devices.

The block cyclically writes the output value of data type DINT for the output of a fail-safe DP standard slave/IO standard device addressed at the VALUE output to the associated F-module driver F_PS_12 that communicates with the fail-safe DP standard slave/IO standard device via a safety message frame according to the PROFIsafe bus profile. The F-module driver is automatically placed and interconnected with the CFC function "Generate module drivers".

A quality code that can have the following states is generated for the output value that is written to the fail-safe DP standard slave/IO standard device:

| State | Quality code (QUALITY output) |
|------------------------|-------------------------------|
| Valid value | 16#80 |
| Simulation | 16#60 |
| Fail-safe value | 16#48 |
| Invalid value (F-STOP) | 16#00 |

I/Os

| | Name | Data type | Explanation | Default |
|-----------------|----------|-----------|---|-------------------------------|
| Inputs: | ADR_CODE | DWORD | Coding for VALUE interconnection | To be automatically supplied* |
| | I | F_DINT | Process value | 0 |
| | SIM_I | F_DINT | Simulation value | 0 |
| | SIM_MOD | F_BOOL | 1=Simulation value takes precedence | 0 |
| | SIM_ON | F_BOOL | 1=Activate simulation value | 0 |
| | PASS_ON | F_BOOL | 1=Enable passivation | 0 |
| | ACK_NEC | F_BOOL | 1=User acknowledgement for reintegration after error required | 0 |
| | ACK_REI | F_BOOL | Acknowledgement for reintegration | 0 |
| | IPAR_EN | F_BOOL | 1=Enable assignment of I-parameters | 0 |
| Outputs: | PASS_OUT | F_BOOL | 1=Passivation due to error | 0 |
| | QBAD | F_BOOL | 1=Process value invalid | 0 |
| | QSIM | F_BOOL | 1=Simulation active | 0 |
| | VALUE | DINT | Address of the output channel | 0 |
| | QUALITY | BYTE | Value status (quality code) of the process value | B#16#0 |
| | ACK_REQ | BOOL | Acknowledgement for reintegration required | 0 |
| | IPAR_OK | F_BOOL | 1= New I-parameter values were assigned | 0 |

*) The ADR_CODE input is automatically supplied when the S7 program is compiled and must not be changed. The ADR_CODE input is displayed as changed during the comparison of safety programs if changes were made to the address or the symbol of the signal at the VALUE input.

Addressing

You must interconnect the symbol generated with *HW Config* in the symbol table for the output value of data type DINT with the VALUE output.

Normal value

The process value active at the I input is written to the fail-safe DP standard slave/IO standard device. The quality code (QUALITY) is set to 16#80.

Simulation

A simulation value can also be written to the fail-safe DP standard slave/IO standard device instead of the process value active at the I input.

When input SIM_ON = 1 and input SIM_MOD = 0, the value of the SIM_I input is written to the fail-safe DP standard slave/IO standard device and output at the VALUE output, if no communication error (PROFIsafe), no module or channel fault (e.g. wire break) and no F-startup is present.

When input SIM_ON = 1 and input SIM_MOD = 1, the value of the SIM_I input is output at the VALUE output even when a communication error (PROFIsafe), module or channel fault (e.g. wire break) or F-startup is present, in order to simulate an "error-free" operation without the presence of a real fail-safe DP standard slave/IO standard device.

In both cases, the quality code (QUALITY) is set to 16#60 and QSIM = 1 is set.

Note

If you have placed more than one F-channel driver for outputs for a fail-safe DP standard slave/IO standard device, a simulation value is not written if the PASS_ON input of another F-channel driver for outputs of the fail-safe DP standard slave/IO standard device is 1 and the SIM_ON input is 0.

Fail-safe value

The fail-safe value 0 is written to the fail-safe DP standard slave/IO standard device when any of the following occurs:

- A communication error (PROFIsafe)
- A module or channel fault (e.g. wire break)
- An F-startup
- Passivation with PASS_ON = 1

The quality code (QUALITY) is set to 16#48 and QBAD = 1 is set.

If the output of the fail-safe value is not caused by a passivation, PASS_OUT = 1 is additionally set for passivation of other channels.

Note

For fail-safe DP standard slaves/IO standard devices, channel-specific passivation is not possible for outputs using PASS_ON. If you have placed more than one F-channel driver for outputs for a fail-safe DP standard slave/IO standard device, the fail-safe value 0 is written for all outputs of the fail-safe DP standard slave/IO standard device following a passivation with PASS_ON = 1 at one of the F-channel drivers. If you want to evaluate the QBAD and QUALITY outputs of the other F-channel drivers when PASS_ON = 1 at one of the F-channel drivers, you must control the PASS_ON inputs of all F-channel drivers synchronously.

Reintegration

After elimination of an error, the fail-safe DP standard slave/IO standard device can be reintegrated automatically or only after user acknowledgment.

With parameter assignment `ACK_NEC = 1`, a user acknowledgement is required at the `ACK_REI` input after error elimination. With parameter assignment `ACK_NEC = 0`, reintegration occurs automatically.

Output `ACK_REQ = 1` is used to signal that the error is eliminated and a user acknowledgment at the `ACK_REI` input is required for reintegration.

No user acknowledgement is required for reintegration after `PASS_ON = 1`. For reintegration after an F-startup following a CPU STOP, no user acknowledgment is required if the fail-safe DP standard slave/IO standard device starts up using the Slave State (20) "system start" according to the PROFIsafe Specification V1.30 and higher. Otherwise, a communication error (PROFIsafe) is detected.

Note

For fail-safe DP standard slaves/IO standard devices, channel-specific reintegration is not possible for outputs using `PASS_ON`. If you have placed more than one F-channel driver for outputs for a fail-safe DP standard slave/IO standard device, you must synchronously control the `ACK_REI` inputs of all F-channel drivers for outputs of the fail-safe DP standard slave/IO standard device.

WARNING

Assignment of input `ACK_NEC = 0` is only allowed if an automatic reintegration is permissible under safety aspects for the process.

Communication errors (PROFIsafe) must always be acknowledged at the `ACK_REI` input independent of `ACK_NEC`. For this purpose, you must interconnect the `ACK_REI` input with a signal generated by an operator input. An interconnection with an automatically generated signal is not permissible.

WARNING

Startup protection for short-term power failure of the fail-safe DP standard slave/IO standard device

After a power failure of the fail-safe DP standard slave/IO standard device that lasts for less time than the F-monitoring time (see section "Run times, F-Monitoring times, and response times (Page 460)") set for the fail-safe DP standard slave/IO standard device in *HW Config*, an automatic reintegration, as described for parameter assignment `ACK_NEC = 0`, may occur irrespective of your parameter assignment of the `ACK_NEC` input.

If automatic reintegration is not permitted for the process involved, you must program a startup protection for this situation by evaluating the `QBAD` or `PASS_OUT` outputs.

At a power failure of the F-I/O lasting longer than the F-monitoring time set for the F-I/O in *HW Config*, the F-system detects a communication error.

Parameter reassignment of a fail-safe DP standard slave/IO standard device

For parameter reassignment of a fail-safe DP standard slave/IO standard device, the IPAR_EN input and the IPAR_OK output are available.

The IPAR_EN input corresponds to the iPar_EN_C variable and the IPAR_OK output corresponds to the iPar_OK_S variable in the PROFIsafe bus profile as of PROFIsafe Specification V1.30. To find out when you must set/reset the IPAR_EN input following parameter reassignment of a fail-safe DP standard slave/IO standard device and how you can evaluate the IPAR_OK output, refer to the PROFIsafe Specification V1.30 or higher and the documentation for the fail-safe standard DP slave/IO standard device.

If more than one F-channel driver is placed for a fail-safe DP standard slave/IO standard device, iPar_EN_C is formed from an OR logic operation of all IPAR_EN of the F-channel drivers belonging to the fail-safe DP standard slave/IO standard device.

If passivation is to be carried out when IPAR_EN = 1, you must additionally set variable PASS_ON = 1.

Startup behavior

After an F-startup, communication must first be established between the F-module driver and the fail-safe DP standard slave/IO standard device. During this time, the fail-safe value 0 is written to the fail-safe DP standard slave/IO standard device. The quality code (QUALITY) is set to 16#48 and outputs QBAD = 1 and PASS_OUT = 1 are set.

Error handling

- An F-STOP is triggered when there is an error in the safety data format in the associated instance DB. The following diagnostics event is then entered in the diagnostics buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

Response to F-STOP

Following an F-STOP, 16#00 is output at the QUALITY output and QBAD.DATA = 1 is set. All other variables are frozen.

A.2.6.12 F_CH_RI: F-channel driver for inputs of data type "REAL" of fail-safe DP standard slaves and fail-safe IO standard devices

Function

The F-block is used for signal processing of an input value of data type REAL of fail-safe DP standard slaves and fail-safe IO standard devices.

The F-block cyclically reads the input value addressed at the VALUE input of data type REAL of a fail-safe DP standard slave/IO standard device from the associated F-module driver F_PS_12 that communicates with the fail-safe DP standard slave/IO standard device via a safety message frame according to the PROFIsafe bus profile. The F-module driver is automatically placed and interconnected with the CFC function "Generate module drivers".

If the input value is valid, it is made available at the V output as Real.

A quality code (QUALITY output) that can have the following states is generated for the result value at the V output:

| State | Quality code (QUALITY output) |
|------------------------|-------------------------------|
| Valid value | 16#80 |
| Simulation | 16#60 |
| Fail-safe value | 16#48 |
| Last valid value | 16#44 |
| Invalid value (F-STOP) | 16#00 |

I/Os

| | Name | Data type | Explanation | Default |
|-----------------|----------|-----------|---|-------------------------------|
| Inputs: | ADR_CODE | DWORD | Coding for VALUE interconnection | to be automatically supplied* |
| | VALUE | REAL | Address of the analog input channel | 0.0 |
| | SIM_V | F_REAL | Simulation value | 0.0 |
| | SIM_ON | F_BOOL | 1 = Activate simulation value | 0 |
| | SUBS_V | F_REAL | Fail-safe value | 0.0 |
| | SUBS_ON | F_BOOL | 1 = Enable fail-safe value injection | 0 |
| | PASS_ON | F_BOOL | 1 = Activate passivation | 0 |
| | ACK_NEC | F_BOOL | 1 = User acknowledgement for reintegration after error required | 0 |
| | ACK_REI | F_BOOL | Acknowledgment for reintegration | 0 |
| | IPAR_EN | F_BOOL | 1 = Enable assignment of I-parameters | 0 |
| Outputs: | PASS_OUT | F_BOOL | 1 = Passivation due to error | 0 |
| | QBAD | F_BOOL | 1 = Process value invalid | 0 |
| | QSIM | F_BOOL | 1 = Simulation active | 0 |
| | QSUBS | F_BOOL | 1 = Fail-safe value injection active | 0 |
| | V | F_REAL | Process value | 0.0 |
| | V_DATA | REAL | DATA component of the process value (for visualization) | 0.0 |
| | QUALITY | BYTE | Value status (quality code) of the process value | 0 |
| | V_MOD | REAL | Value from fail-safe DP standard slave/IO standard device | 0.0 |
| | ACK_REQ | BOOL | Acknowledgment for reintegration required | 0 |
| | IPAR_OK | F_BOOL | 1= New I-parameter values were assigned | 0 |

*) The ADR_CODE input is automatically supplied when the S7 program is compiled and must not be changed. The ADR_CODE input is displayed as changed during the comparison of safety programs if changes were made to the address or the symbol of the signal at the VALUE input.

Note**Forcing the VALUE input**

Forcing of the VALUE input is not possible because VALUE is not the value itself but rather a pointer for the address of the digital input channel.

Addressing

You must interconnect the symbol generated with *HW Config* in the symbol table for the input value of data type REAL with the VALUE input.

Normal value

If the input value received from the fail-safe DP standard slave/IO standard device is valid, it is output at the Q output with quality code (QUALITY) 16#80.

Simulation

A simulation value is output at the V output instead of the normal value that is received from the fail-safe DP standard slave/IO standard device.

When input SIM_ON = 1, the value of the SIM_V input with quality code (QUALITY) 16#60 is output. Simulation has the highest priority. QBAD is always set to "0". QSIM = 1 is set if the block is in simulation state.

When simulation is switched on, the input value received from the fail-safe DP standard slave/IO standard device is output at the V_MOD output. If no communication is possible with the fail-safe DP standard slave/IO standard device or if there is still no user acknowledgement after an error, 0 is output.

When simulation is switched off, V_DATA is output.

Fail-safe value

When input SUBS_ON = 1, the fail-safe value SUBS_V is output at the V output in the following cases:

- The analog input value is invalid due to a communication error (PROFIsafe).
- The analog input value is invalid due to a module fault or a fail-safe value is received from the module.
- A passivation with PASS_ON = 1 is present.
- An F-startup is present.
- FV_ACTIVATED is signaled by the module.

The quality code (QUALITY) is set to 16#48 and QSUBS = 1 and QBAD = 1 are set.

If the output of the fail-safe value is not caused by a passivation, PASS_OUT = 1 is additionally set for passivation of other channels.

Keep last value

When input SUBS_ON = 0, the last valid value of V is output at the V output in the following cases:

- The analog input value is invalid due to a communication error (PROFIsafe).
- The analog input value is invalid due to a module fault or a fail-safe value is received from the module.
- A passivation with PASS_ON = 1 is present.
- FV_ACTIVATED is signaled by the module.

The quality code (QUALITY) is set to 16#44 and QSUBS = 0 and QBAD = 1 are set.

If the output of the last valid value is not caused by a passivation, then output PASS_OUT = 1 is additionally set in order to passivate other channels.

Parameter reassignment of a fail-safe DP standard slave/IO standard device

For parameter reassignment of a fail-safe DP standard slave/IO standard device, the IPAR_EN input and the IPAR_OK output are available.

The IPAR_EN input corresponds to the iPar_EN_C variable and the IPAR_OK output corresponds to the iPar_OK_S variable in the PROFIsafe bus profile as of PROFIsafe Specification V1.30. To find out when you must set/reset the IPAR_EN input following parameter reassignment of a fail-safe DP standard slave/IO standard device and how you can evaluate the IPAR_OK output, refer to the PROFIsafe Specification V1.30 or higher and the documentation for the fail-safe standard DP slave/IO standard device.

If more than one F-channel driver is placed for a fail-safe DP standard slave/IO standard device, iPar_EN_C is formed from an OR logic operation of all IPAR_EN of the F-channel driver belonging to the fail-safe DP standard slave/IO standard device.

If passivation is to be carried out when IPAR_EN = 1, you must additionally set variable PASS_ON = 1.

Reintegration after error elimination

After elimination of an error, the input value received from the fail-safe DP standard slave/IO standard device can be reintegrated automatically or only after user acknowledgment.

With parameter assignment ACK_NEC = 1, a user acknowledgement is required at the ACK_REI input after error elimination. With parameter assignment ACK_NEC = 0, reintegration occurs automatically.

Output ACK_REQ = 1 is used to signal that the error is eliminated and a user acknowledgment at the ACK_REI input is required for reintegration.

No user acknowledgement is required for reintegration after PASS_ON = 1. For reintegration after an F-startup following a CPU STOP, no user acknowledgement is required if the fail-safe DP standard slave/IO standard device starts up using the Slave State (20) "system start"

according to the PROFIsafe Specification V1.30 and higher. Otherwise, a communication error (PROFIsafe) is detected.

WARNING

Parameter assignment of input ACK_NEC = 0 is only allowed if an automatic reintegration is permissible under safety aspects for the process.

Communication errors (PROFIsafe) must always be acknowledged at the ACK_REI input independent of ACK_NEC. For this purpose, you must interconnect the ACK_REI input with a signal generated by an operator input. An interconnection with an automatically generated signal is not permissible.

WARNING

Startup protection for short-term power failure of the fail-safe DP standard slave/IO standard device

After a power failure of the fail-safe DP standard slave/IO standard device that lasts for less time than the F-monitoring time (see section "Run times, F-Monitoring times, and response times (Page 460)") set for the fail-safe DP standard slaves/IO standard devices in *HW Config*, an automatic reintegration, as described for parameter assignment ACK_NEC = 0, may occur irrespective of your parameter assignment of the ACK_NEC input.

If automatic reintegration is not permitted for the process involved, you must program a startup protection for this situation by evaluating the QBAD or PASS_OUT outputs.

At a power failure of the fail-safe DP standard slave/IO standard device lasting longer than the F-monitoring time set for the fail-safe DP standard slave/IO standard device in *HW Config*, the F-system detects a communication error.

Startup behavior

After an F-startup, communication must first be established between the F-module driver and the fail-safe DP standard slave/IO standard device. During this time, the fail-safe value SUBS_V with quality code (QUALITY output) 16#48 is output irrespective of the parameter assignment at the SUBS_ON input and outputs QSUBS = 1, QBAD = 1 and PASS_OUT = 1 are additionally set.

Error handling

An F-STOP is triggered when there is an error in the safety data format in the associated instance DB. The following diagnostics event is then entered in the diagnostics buffer of the F-CPU:

- "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

Response to F-STOP

Following an F-STOP, 16#00 is output at the QUALITY and STATUS outputs and QBAD.DATA = 1 is set. All other variables are frozen.

A.2.7 F-System blocks

Overview

| Block name | Block number | Description |
|------------|--------------|---|
| F_S_BO | FB 390 | Sending of 10 data elements of data type F_BOOL in a fail-safe manner to another F-Shutdown group |
| F_R_BO | FB 391 | Receiving of 10 data elements of data type F_BOOL in a fail-safe manner from another F-Shutdown group |
| F_S_R | FB 392 | Sending of 5 data elements of data type F_REAL in a fail-safe manner to another F-Shutdown group |
| F_R_R | FB 393 | Receiving of 5 data elements of data type F_REAL in a fail-safe manner from another F-Shutdown group |
| F_START | FB 394 | F-Start detection |
| F_PSG_M | FB 471 | Marker block for F-Shutdown groups |

Integration in F Block types

With the exception of F_START, the F-System blocks must not be integrated in F-Block types.

A.2.7.1 F_S_BO: Sending of 10 data elements of data type F_BOOL in a fail-safe manner to another F-Shutdown group.

Function

The F-Block transfers the data of data type F_BOOL fail-safe adjacent to input SD_BO_xx to another F-Shutdown group. The data must be received there with the F_R_BO F-Block.

You must interconnect output S_DB with the input of the same name of the corresponding F_R_BO.

Note

Initialization

You are not allowed to initialize output S_DB with values <> 0.

Inputs/outputs

| | Name | Data type | Description | Default |
|---------|----------|-----------|-------------------|---------|
| Inputs: | SD_BO_00 | F_BOOL | SEND DATA BOOL 00 | 0 |
| | ... | | ... | |
| | SD_BO_09 | F_BOOL | SEND DATA BOOL 09 | 0 |

| | Name | Data type | Description | Default |
|----------------|------|-----------|----------------------|---------|
| Output: | S_DB | F_WORD | Connection to F_R_BO | 0 |

Error handling

None

A.2.7.2 F_R_BO: Receiving of 10 data elements of data type F_BOOL in a fail-safe manner from another F-Shutdown group

Function

This F-Block receives 10 data elements of data type F_BOOL fail-safe from another F-Shutdown group and makes them available on outputs RD_BO_xx. The data must be transferred from the other F-Shutdown group with the F_S_BO F-Block. Interconnect the data at outputs RD_BO_xx for further processing with other F-Blocks.

You must interconnect input S_DB with the output of the same name of the corresponding F_S_BO.

You must assign the desired F-Monitoring time at input TIMEOUT. For information about calculating F-Monitoring times, refer to chapter " Run times, F-Monitoring times, and response times (Page 460) ".

Inputs/outputs

| | Name | Data type | Description | Default |
|-----------------|----------|-----------|--------------------------|---------|
| Inputs: | TIMEOUT | F_TIME | F-MONITORING TIME | T# 0ms |
| | S_DB | F_WORD | Connection to F_S_BO | 0 |
| | SUBBO_00 | F_BOOL | SUBSTITUTE FOR BOOL 00 | 0 |
| | ... | | ... | |
| | SUBBO_09 | F_BOOL | SUBSTITUTE FOR BOOL 09 | 0 |
| Outputs: | SUBS_ON | F_BOOL | 1 = SUBSTITUTE OUTPUT ON | 0 |
| | RD_BO_00 | F_BOOL | RECEIVED DATA BOOL 00 | 0 |
| | ... | | ... | |
| | RD_BO_09 | F_BOOL | RECEIVED DATA BOOL 09 | 0 |

Substitute values

In the following cases the configured substitute values at inputs SUBBO_xx are output at outputs RD_BO_xx:

- No updated data is received from the corresponding F_S_BO within the configured F-Monitoring time at input TIMEOUT, because for example partial shutdown is pending for the F-Shutdown group with the corresponding F_S_BO.
- An F-Startup is pending.

The SUBS_ON output is set to 1.

Startup characteristics

After an F-Startup data exchange has to first be established with the corresponding F_S_BO. In this case the configured substitute values at inputs SUBBO_XX are output at outputs RD_BO_XX and output SUBS_ON is set to 1.

Error handling

An F_STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:

- "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.7.3 F_S_R: Sending of 5 data elements of data type F_REAL in a fail-safe manner to another F-Shutdown group

Function

This F-Block transfers the data of data type F_REAL fail-safe from the input SD_R_XX to another F-Shutdown group. The data must be received there with the F_R_R F-Block.

You must interconnect output S_DB with the input of the same name of the corresponding F_R_R.

Note

Initialization

You are not allowed to initialize output S_DB with values $\neq 0$.

Inputs/outputs

| | Name | Data type | Description | Default |
|---------|---------|-----------|---------------------|---------|
| Inputs: | SD_R_00 | F_REAL | SEND DATA REAL 00 | 0.0 |
| | ... | | ... | |
| | SD_R_04 | F_REAL | SEND DATA REAL 04 | 0.0 |
| Output: | S_DB | F_WORD | Connection to F_R_R | 0 |

Error handling

None

A.2.7.4 F_R_R: Receiving of 5 data elements of data type F_REAL in a fail-safe manner from another F-Shutdown group.

Function

This F-Block receives 5 data elements of data type F_REAL fail-safe from another F-Shutdown group and makes them available on outputs RD_BO_xx. The data must be transferred from the other F-Shutdown group with the F_S_R F-Block.

You must interconnect input S_DB with the output of the same name of the corresponding F_S_R.

You must assign the desired F-Monitoring time at input TIMEOUT. For information about calculating F-Monitoring times, refer to chapter " Run times, F-Monitoring times, and response times (Page 460) ".

Inputs/outputs

| | Name | Data type | Description | Default |
|-----------------|---------|-----------|---------------------------------|---------|
| Inputs: | TIMEOUT | F_TIME | F-MONITORING TIME | T# 0ms |
| | S_DB | F_WORD | Connection to F_S_R | 0 |
| | SUBR_00 | F_REAL | SUBSTITUTE FOR REAL 00 | 0.0 |
| | ... | | ... | |
| | SUBR_04 | F_REAL | SUBSTITUTE FOR REAL 04 | 0.0 |
| Outputs: | SUBS_ON | F_BOOL | 1 = Fail-safe values are output | 0 |
| | RD_R_00 | F_REAL | RECEIVED REAL 00 | 0.0 |
| | ... | | ... | |
| | RD_R_04 | F_REAL | RECEIVED REAL 04 | 0.0 |

Substitute values

In the following cases the configured substitute values at inputs SUBR_xx are output at outputs RD_R_xx:

- No updated data is received from the corresponding F_S_R within the configured F-Monitoring time at input TIMEOUT, because for example partial shutdown is pending for the F-Shutdown group with the corresponding F_S_R.
- An F-Startup is pending.

The SUBS_ON output is set to 1.

Startup characteristics

The data exchange has to first be established with the corresponding F_S_R following an F-Startup. At this time the configured substitute values at inputs SUBR_xx are output at outputs RD_R_xx and the output SUBS_ON is set to 1.

Error handling

An F_STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:

- "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.7.5 F_START: F-Startup identifier

Function

In the first cycle after an F-Startup or a initial run, the F-Block signals with 1 at output COLDSTRT that an F-Startup was executed. COLDSTRT remains present until the next call of F_START.

The F_START must be called before the evaluating F-Blocks.

Inputs/outputs

| | Name | Data type | Description | Default |
|----------------|----------|-----------|----------------------|---------|
| Output: | COLDSTRT | F_BOOL | F-Startup identifier | 1 |

Error handling

None

A.2.7.6 F_PSG_M: Marker block for F-Shutdown groups

Function

With the F_PSG_M block you have the possibility to split an F-Shutdown group into two F-Shutdown groups.

In the sequence editor of the CFC editor, place the block F_PSG_M in the last F-Runtime group, which should belong to the first F-Shutdown group. Any following F-Runtime groups then form the second F-Shutdown group. The F_PSG_M block is not an F-Block. However, you are still permitted to place it in F-Runtime groups.

Inputs/outputs:

None

Error handling:

None

See also

F-Shutdown groups (Page 84)

A.2.8 Flip-flop blocks**Overview**

| Block name | Block number | Description |
|------------|--------------|----------------------------------|
| F_RS_FF | FB 307 | RS Flip-Flop, resetting dominant |
| F_SR_FF | FB 308 | SR Flip-Flop, setting dominant |

A.2.8.1 F_RS_FF: RS Flip-Flop, resetting dominant**Function**

This F-Block executes the function of an RS Flip-Flops (resetting dominant). The output Q is set when input R = 0 and input S = 1. The output Q is reset when input R = 1 and input S = 0. Output Q is reset if 1 is at both inputs. The QN output corresponds to the negated Q output.

Truth table

| R | S | Qn | QNn |
|---|---|------|-------|
| 0 | 0 | Qn-1 | QNn-1 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 |
| 1 | 1 | 0 | 1 |

Inputs/outputs

| | Name | Data type | Description | Default |
|-----------------|------|-----------|----------------|---------|
| Inputs: | R | F_BOOL | Reset | 0 |
| | S | F_BOOL | Set | 0 |
| Outputs: | Q | F_BOOL | Output | 0 |
| | QN | F_BOOL | Negated output | 1 |

Error handling

An F_STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:

- "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.8.2 F_SR_FF: SR Flip-Flop, setting dominant

Function

The F-Block executes the function of an *SR Flip-Flop* (setting dominant). The output Q is set when input R = 0 and input S = 1. The output Q is reset when input R = 1 and input S = 0. Output Q is set if 1 is at both inputs. The QN output corresponds to the negated Q output.

Truth table

| R | S | Qn | QNn |
|---|---|------|-------|
| 0 | 0 | Qn-1 | QNn-1 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 0 |

Inputs/outputs

| | Name | Data type | Description | Default |
|-----------------|------|-----------|----------------|---------|
| Inputs: | R | F_BOOL | Reset | 0 |
| | S | F_BOOL | Set | 0 |
| Outputs: | Q | F_BOOL | Output | 0 |
| | QN | F_BOOL | Negated output | 1 |

Error handling

An F_STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:

- "Safety program: Error in safety data format in DB" (Event ID16#75DA).

A.2.9 IEC pulse and counter blocks

Overview

| Block name | Block number | Description |
|------------|--------------|------------------------|
| F_CTUD | FB 341 | Up and down counter |
| F_TP | FB 342 | Timer pulse |
| F_TON | FB 343 | Timer switch-on delay |
| F_TOF | FB 344 | Timer switch-off delay |

A.2.9.1 F_CTUD: Up and down counter

Function

This F-Block is an edge-controlled up/down counter.

The CV count value responds to rising edges of the CU and CD inputs as well as to the level of the LOAD and R inputs:

- Rising edge at CU: CV is increased by 1.
When the counter value reaches the upper limit (32.767), it no longer counts up.
- Rising edge at CD: CV is decreased by 1.
When the counter value reaches the lower limit (-32.768), it no longer counts down.
- LOAD = 1: CV is preset with the value of the PV input.
The values at inputs CU and CD are ignored.
- R = 1: CV is reset to 0.
The values at inputs CU, CD, and LOAD are ignored.

If a rising edge is available at both the CU input and the CD input during a cycle, the counter keeps its current value.

The QU output is set if the count value is greater than or equal to the preset value PV. The QD output is set if the count value is less than or equal to zero.

Inputs/outputs

| | Name | Data type | Description | Default |
|----------------|------|-----------|--------------|---------|
| Inputs: | CU | F_BOOL | COUNT UP | 0 |
| | CD | F_BOOL | COUNT DOWN | 0 |
| | R | F_BOOL | RESET | 0 |
| | LOAD | F_BOOL | LOAD PV | 0 |
| | PV | F_INT | PRESET VALUE | 0 |

| | Name | Data type | Description | Default |
|-----------------|------|-----------|--|---------|
| Outputs: | QU | F_BOOL | COUNTER UP QU has the value <ul style="list-style-type: none"> • 1: If CV ≥ PV • 0: If CV < PV | 0 |
| | QD | F_BOOL | COUNTER DOWN QD has the value <ul style="list-style-type: none"> • 1: If CV ≤ 0 • 0: If CV > 0 | 0 |
| | CV | F_INT | COUNTER VALUE | 0 |

Error handling

An F_STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:

- "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.9.2 F_TP: Timer pulse

Function

The F-Block generates a pulse with duration PT at output Q.

The pulse is initiated on a rising edge at input IN. Output Q remains set for duration PT, irrespective of any further variation of the input signal (that is, even if input IN switches from 0 back to 1 before time PT has elapsed).

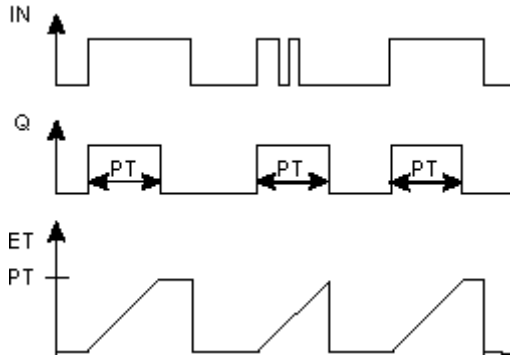
Output ET displays how long output Q has already been set. It can have a maximum value equal to the value of input PT. It is reset when the input IN changes to 0, but only after time PT expires.

If PT < 0, outputs Q and ET are reset.


Inputs/outputs

| | Name | Data type | Description | Default |
|-----------------|------|-----------|--------------|---------|
| Inputs: | IN | F_BOOL | START INPUT | 0 |
| | PT | F_TIME | TIMESSET | T# 0ms |
| Outputs: | Q | F_BOOL | OUTPUT | 0 |
| | ET | F_TIME | ELAPSED TIME | T# 0ms |

Timing diagram



Fail-safe user times

| |
|---|
|  WARNING |
| <p>When using an F-Block with time processing, take the following timing imprecision sources into account when determining your response times:</p> <ul style="list-style-type: none"> • Known timing imprecision (based on standard systems) resulting from cyclic processing • Tolerance of internal time monitoring in the F-CPU <ul style="list-style-type: none"> – For time values of 10 ms to 50 s: 5 ms – For time values of $> n \times 50 \text{ s}$ to $(n+1) \times 50 \text{ s}$: $\pm (n+1) \times 5 \text{ ms}$ |

Error handling

An F_STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:

- "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.9.3 F_TON: Timer switch-on delay

Function

This F-Block delays a rising edge by the time PT.

A rising edge at input IN results in a rising edge at output Q once time PT has elapsed. Q remains set until input IN changes to 0.

If input IN changes back to 0 before time PT has elapsed, then output Q remains at 0.

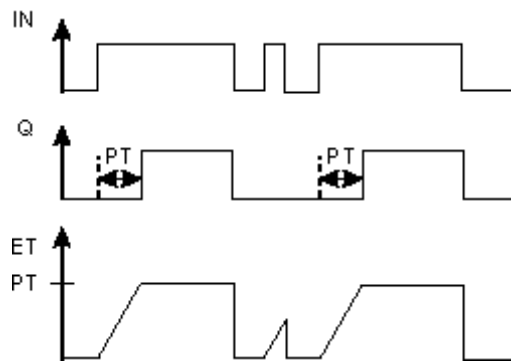
Output ET supplies the time that has passed since the last rising edge at input IN, not to exceed the value at input PT. ET is reset if input IN changes to 0.

If $PT < 0$, outputs Q and ET are reset.


Inputs/outputs

| | Name | Data type | Description | Default |
|----------|------|-----------|--------------|---------|
| Inputs: | IN | F_BOOL | START INPUT | 0 |
| | PT | F_TIME | TIMESET | T# 0ms |
| Outputs: | Q | F_BOOL | OUTPUT | 0 |
| | ET | F_TIME | ELAPSED TIME | T# 0ms |

Timing diagram



Fail-safe user times

|  WARNING |
|---|
| <p>When using an F-Block with time processing, take the following timing imprecision sources into account when determining your response times:</p> <ul style="list-style-type: none"> • Known timing imprecision (based on standard systems) resulting from cyclic processing • Tolerance of internal time monitoring in the F-CPU <ul style="list-style-type: none"> – For time values of 10 ms to 50 s: 5 ms – For time values of $> n \times 50 \text{ s}$ to $(n+1) \times 50 \text{ s}$: $\pm (n+1) \times 5 \text{ ms}$ |

Error handling

An F_STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:

- "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.9.4 F_TOF: Timer switch-off delay

Function

This F-Block delays a falling edge by the time PT.

A rising edge at input IN causes a rising edge at output Q. A falling edge at input IN results in a falling edge at output Q once time PT has elapsed.

If input IN changes back to 1 before time PT has elapsed, then output Q remains at 1.

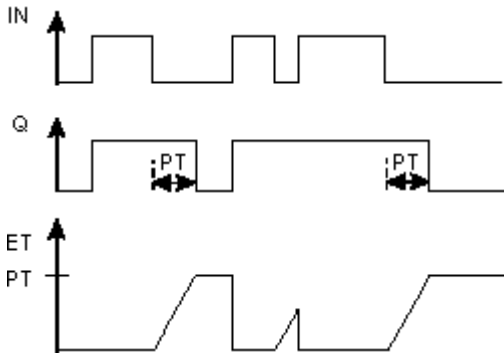
Output ET supplies the time that has passed since the last falling edge at input IN, not to exceed the value at input PT. ET is reset if input IN changes to 1.

If $PT < 0$, output ET is reset and output Q corresponds to input IN.


Inputs/outputs

| | Name | Data type | Description | Default |
|----------|------|-----------|--------------|---------|
| Inputs: | IN | F_BOOL | START INPUT | 0 |
| | PT | F_TIME | TIMESSET | T# 0ms |
| Outputs: | Q | F_BOOL | OUTPUT | 0 |
| | ET | F_TIME | ELAPSED TIME | T# 0ms |

Timing diagram



Fail-safe user times

|  WARNING |
|---|
| <p>When using an F-Block with time processing, take the following timing imprecision sources into account when determining your response times:</p> <ul style="list-style-type: none"> • Known timing imprecision (based on standard systems) resulting from cyclic processing • Tolerance of internal time monitoring in the F-CPU <ul style="list-style-type: none"> – For time values of 10 ms to 50 s: 5 ms – For time values of $> n \times 50$ s to $(n+1) \times 50$ s: $\pm (n+1) \times 5$ ms |

Error handling

An F_STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:

- "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.10 Pulse blocks

Overview

| Block name | Block number | Description |
|------------|--------------|---------------------------------------|
| F_REPCYC | FB 309 | Clock |
| F_ROT | FB 310 | Timer with on delay and hold function |
| F_LIM_TI | FB 345 | Asymmetrical limiter of a TIME value |
| F_R_TRIG | FB 346 | Detection of a rising edge |
| F_F_TRIG | FB 347 | Detection of a falling edge |

A.2.10.1 F_REPCYC: Clock

Function

This F-Block implements a clock with an adjustable period, pulse duration, and interpulse period.

A rising edge at input IN starts the clock. The clock starts at output Q with "0" or "1" depending on the setting at input START.

- When input START = 0, the clock first outputs "0" at output Q for the interpulse period, and then "1" for the pulse duration.
- When input START = 1, the clock first outputs "1" at output Q for the pulse duration, and then "0" for the interpulse period.

The clock is repeatedly changed to 0 until IN. Then, Q = 0 is set.

Output ET always supplies the time that has elapsed since the start of a new period. Output RT always supplies the time remaining until the end of the period. ET is reset when a period ends or when IN = 0. RT is set to the period when a period ends or when IN = 0.

Period, pulse duration, and interpulse period are dependent on the settings at the OFFTIME, ONTIME, and PCTON inputs (where $0 \leq \text{PCTON} \leq 100$). OFFTIME, ONTIME, and PCTON must be specified in such a way that the period does not exceed the maximum value of data type TIME.

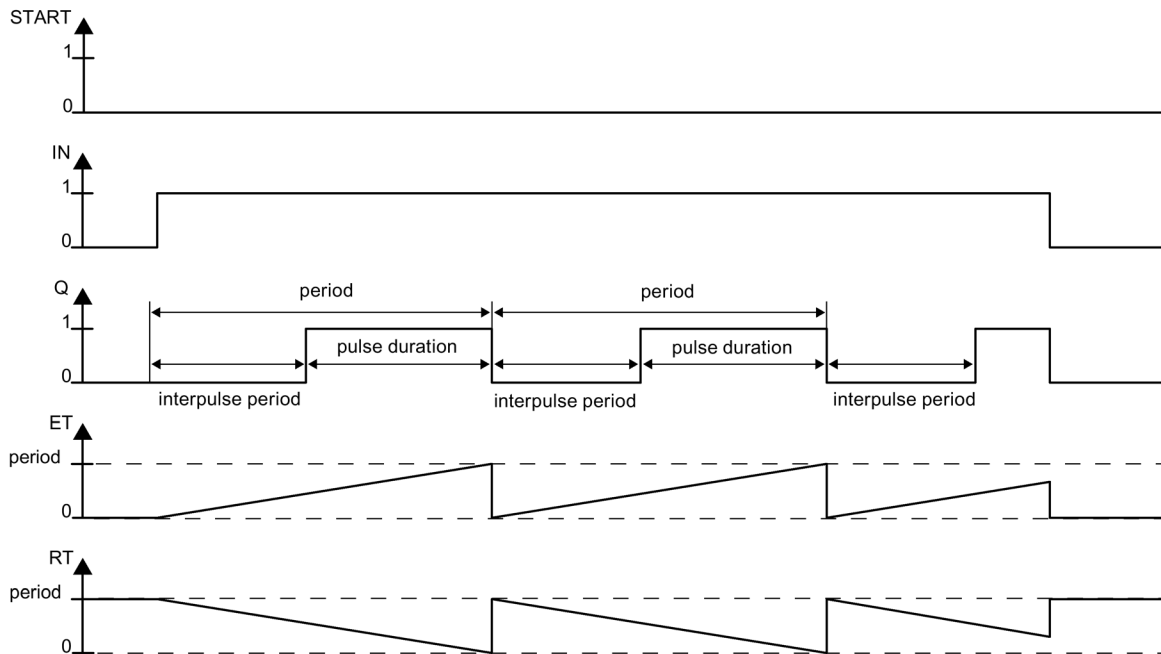
- For OFFTIME > 0 ms, the following applies:
 - Interpulse period = OFFTIME
 - Pulse duration = PCTON x ONTIME
 - Period = OFFTIME + (PCTON x ONTIME)
- For OFFTIME = 0 ms, the following applies:
 - Interpulse period = ONTIME - (PCTON x ONTIME)
 - Pulse duration = PCTON x ONTIME
 - Period = ONTIME

While input IN = 1, the time values at inputs ONTIME and OFFTIME must not be changed.

Inputs/outputs

| | Name | Data type | Description | Default |
|-----------------|---------|-----------|--|---------|
| Inputs: | IN | F_BOOL | Start input | 0 |
| | PCTON | F_REAL | Percentage value for pulse duration | 0 |
| | START | F_BOOL | 0 = Start of period with Q=0 1 = Start of period with Q=1 | 1 |
| | OFFTIME | F_TIME | Parameter for interpulse period | 0 ms |
| | ONTIME | F_TIME | Parameter for pulse duration | 0 ms |
| Outputs: | Q | F_BOOL | Output | 0 |
| | ET | F_TIME | Elapsed time | 0 ms |
| | RT | F_TIME | Remaining time | 0 ms |

Timing diagram



Fail-safe user times

WARNING

When using an F-Block with time processing, take the following timing imprecision sources into account when determining your response times:

- Known timing imprecision (based on standard systems) resulting from cyclic processing
- Tolerance of internal time monitoring in the F-CPU
 - For time values of 10 ms to 50 s: 5 ms
 - For time values of $> n \times 50$ s to $(n+1) \times 50$ s: $\pm (n+1) \times 5$ ms

Error handling

- If input PCTON is an invalid floating-point number (NaN) or a negative time is present at inputs ONTIME or OFFTIME, the clock shuts down (behavior same as when IN = 0). If an invalid floating-point number (NaN) or a negative time is no longer pending and IN = 1, the clock is restarted (behavior same as for a rising edge at input IN).
- When $PCTON < 0.0$, ET and RT are generated same as when PCTON = 0, and Q is set to 0. When $PCTON > 100.0$, ET and RT are generated same as when PCTON = 100, and Q is set to 1.

- If the period exceeds the maximum value of data type TIME, the behavior of the F-Block is undefined.
- An F-STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.10.2 F_ROT: Timer with on delay and hold function

Function

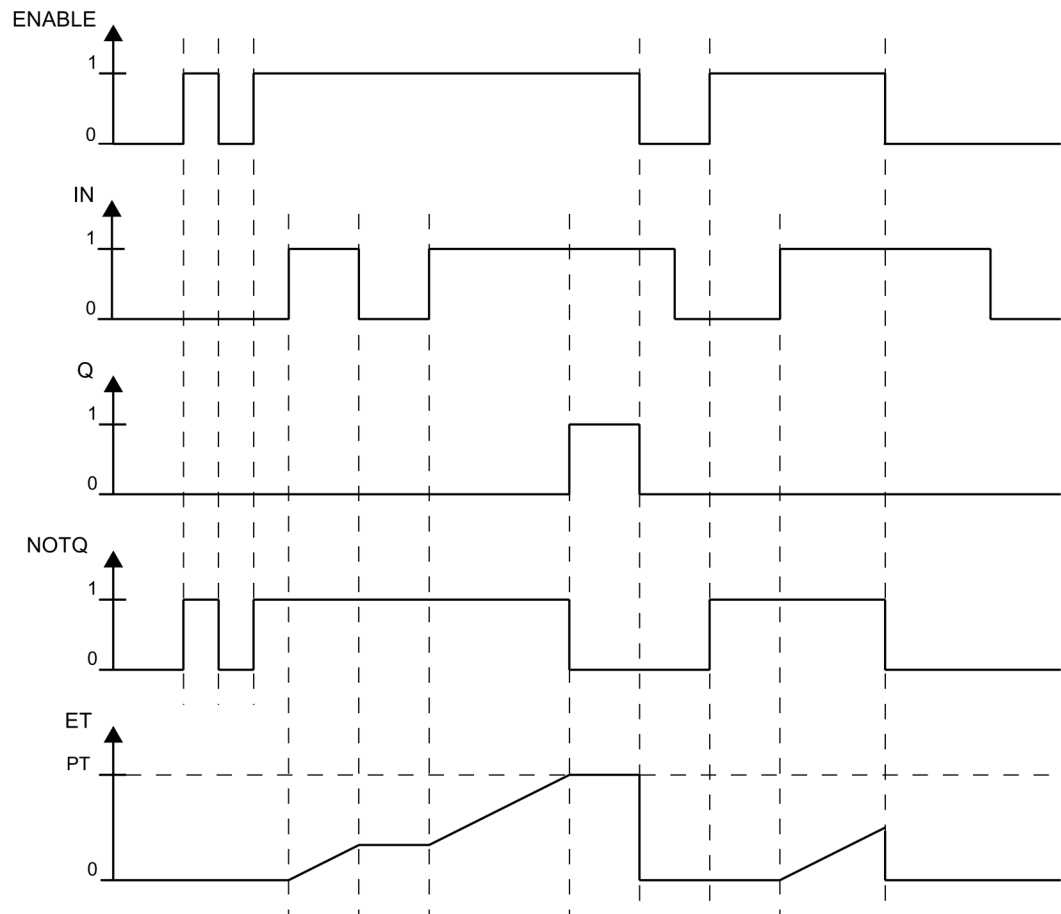
This F-Block implements a timer with on delay and hold function.

- The timer is enabled with input ENABLE = 1. If input IN = 1, the time at output ET is incremented, but only as high as the value of input PT. If IN changes to "0", the time is halted.
 Q is set to "1" as soon as ET = PT. NOTQ corresponds to the inverted Q.
- The timer is reset with input ENABLE = 0. Output ET is set to 0 ms and Q and NOTQ are set to 0.

Inputs/outputs

| | Name | Data type | Description | Default |
|-----------------|--------|-----------|----------------------------------|---------|
| Inputs: | ENABLE | F_BOOL | 1=Enable timer | 0 |
| | IN | F_BOOL | Start input | 0 |
| | PT | F_TIME | Time duration | 0 ms |
| Outputs: | Q | F_BOOL | Output | 0 |
| | NOTQ | F_BOOL | NEGATING OUTPUT (if ENABLE=1) | 0 |
| | ET | F_TIME | Elapsed time | 0 ms |

Timing diagram



Fail-safe user times

WARNING

When using an F-Block with time processing, take the following timing imprecision sources into account when determining your response times:

- Known timing imprecision (based on standard systems) resulting from cyclic processing
- Tolerance of internal time monitoring in the F-CPU
 - For time values of 10 ms to 50 s: 5 ms
 - For time values of $> n \times 50 \text{ s}$ to $(n+1) \times 50 \text{ s}$: $\pm (n+1) \times 5 \text{ ms}$

Error handling

- If a negative time is pending at input PT, the timer is halted (behavior same as when IN = 0). If a negative time is no longer pending, and IN = 1, the timer resumes.
- An F-STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.10.3 F_LIM_TI: Asymmetrical limiter of a TIME value

Function

This F-Block checks whether input IN is within or outside the interval between MIN and MAX. If input IN lies within the interval, it is passed through to output OUT. If it lies outside of the interval it is limited to MIN or MAX.

- Is $IN > MAX$, then an upper limit violation exists. MAX is output to output OUT. OUTU is set to 1 and OUTL to 0.
- If $IN < MIN$, then a lower limit violation exists. MIN is output to output OUT. OUT is set to 0 and OUTL to 1.
- If input IN lies between MIN and MAX, IN is passed through to output OUT. OUTU and OUTL are always set to 0.

Inputs/outputs

| | Name | Data type | Description | Default |
|-----------------|------|-----------|-------------|--------------------------|
| Inputs: | IN | F_TIME | INPUT | T# 0ms |
| | MIN | F_TIME | MINIMUM | T# 0ms |
| | MAX | F_TIME | MAXIMUM | T# 24d 20h 31m 23s 647ms |
| Outputs: | OUT | F_TIME | Output | T# 0ms |
| | OUTU | F_BOOL | UPPER LIMIT | 0 |
| | OUTL | F_BOOL | LOWER LIMIT | 0 |

Error handling

- Is $MIN \geq MAX$, MAX is output at output OUT. OUTU and OUTL are always set to 1.
- An F_STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.10.4 F_R_TRIG: Detection of a rising edge

Function

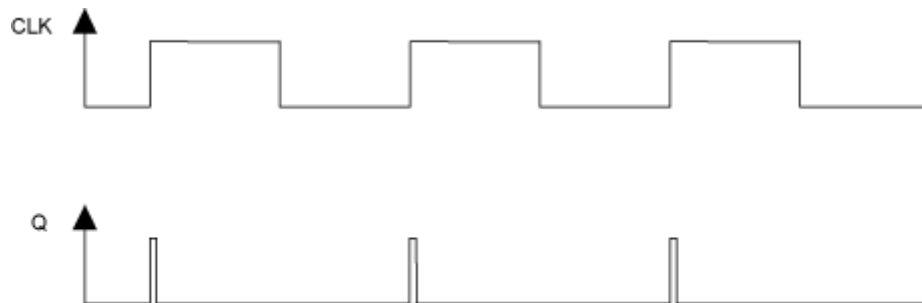
The F-Block checks input CLK for the occurrence of a rising edge.

At a rising edge of input CLK, output Q is set to 1 until the next call of the block.

Inputs/outputs

| | Name | Data type | Description | Default |
|----------------|------|-----------|-------------|---------|
| Input: | CLK | F_BOOL | Input | 0 |
| Output: | Q | F_BOOL | Output | 0 |

Timing diagram



Startup characteristics

If input CLK has a value of 1 during the first cycle after a F-Startup or an initial run 1, no edge is detected and output Q is set to 0 until the next rising edge on output CLK.

Error handling

None

A.2.10.5 F_F_TRIG: Detection of a falling edge

Function

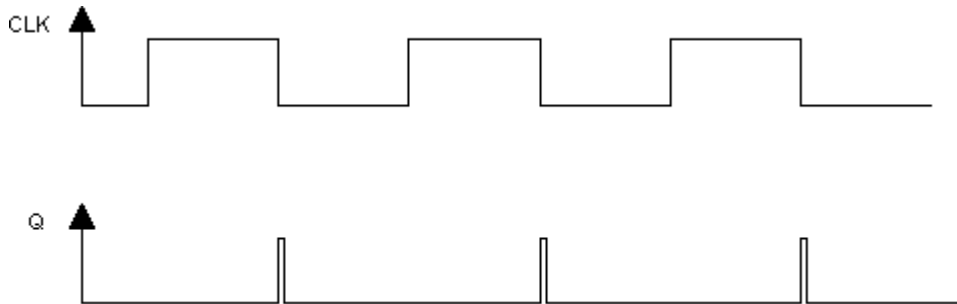
This F-Block checks input CLK for the occurrence of a falling edge.

At a falling edge of input CLK, output Q is set to 1 until the next call of the block.

Inputs/outputs

| | Name | Data type | Description | Default |
|---------|------|-----------|-------------|---------|
| Input: | CLK | F_BOOL | Input | 0 |
| Output: | Q | F_BOOL | Output | 0 |

Timing diagram



Startup characteristics

During the first cycle after a F-Start or initial run, no edge is detected.

Error handling

None

A.2.11 Arithmetic blocks with the REAL data type

Overview

| Block name | Block number | Description |
|------------|--------------|---|
| F_ADD_R | FB 321 | Addition of two REAL values |
| F_SUB_R | FB 322 | Subtraction of two REAL values |
| F_MUL_R | FB 323 | Multiplication of two REAL values |
| F_DIV_R | FB 324 | Division of two REAL values |
| F_ABS_R | FB 325 | Absolute value of a REAL value |
| F_MAX3_R | FB 326 | Maximum of three REAL values |
| F_MID3_R | FB 327 | Mean value of three REAL values |
| F_MIN3_R | FB 328 | Minimum of three REAL values |
| F_LIM_R | FB 329 | Asymmetrical limiter of a REAL value |
| F_SQRT | FB 330 | Square root of a REAL value |
| F_AVEX_R | FB 331 | Mean value of a maximum of nine REAL values |

| Block name | Block number | Description |
|------------|--------------|--|
| F_SMP_AV | FB 333 | Sliding mean value of maximum 33 REAL values |
| F_2oo3_R | FB 456 | Median value of three REAL values with 2oo3 evaluation |
| F_1oo2_R | FB 457 | 1oo2 evaluation of inputs of data type REAL |

A.2.11.1 F_ADD_R: Addition of two REAL values

Function

This F-Block adds the inputs IN1 and IN2 and outputs the sum at output OUT.

$$\text{OUT} = \text{IN1} + \text{IN2}$$

Inputs/outputs

| | Name | Data type | Description | Default |
|---------|------|-----------|-------------|---------|
| Inputs: | IN1 | F_REAL | Input 1 | 0.0 |
| | IN2 | F_REAL | Input 2 | 0.0 |
| Output: | OUT | F_REAL | Output | 0.0 |

Error handling

If an invalid floating-point number (NaN) has been created due to the calculation at output OUT, the following diagnostic event is entered in the diagnostic buffer of the F-CPU:

- "Safety program: invalid REAL number in DB" (Event ID 16#75D9)

A.2.11.2 F_SUB_R: Subtraction of two REAL values

Function

This F-Block subtracts the IN2 input from the IN1 input and outputs the difference at the output OUT.

$$\text{OUT} = \text{IN1} - \text{IN2}$$

Inputs/outputs

| | Name | Data type | Description | Default |
|---------|------|-----------|-------------|---------|
| Inputs: | IN1 | F_REAL | Input 1 | 0.0 |
| | IN2 | F_REAL | Input 2 | 0.0 |
| Output: | OUT | F_REAL | Output | 0.0 |

Error handling

If an invalid floating-point number (NaN) has been created due to the calculation at output OUT, the following diagnostic event is entered in the diagnostic buffer of the F-CPU:

- "Safety program: invalid REAL number in DB" (Event ID 16#75D9)

A.2.11.3 F_MUL_R: Multiplication of two REAL values

Function

This F-Block multiplies the inputs IN1 and IN2 and outputs the product at output OUT.

$$OUT = IN1 \times IN2$$

Inputs/Outputs

| | Name | Data type | Description | Default |
|---------|------|-----------|-------------|---------|
| Inputs: | IN1 | F_REAL | Input 1 | 0.0 |
| | IN2 | F_REAL | Input 2 | 0.0 |
| Output: | OUT | F_REAL | Output | 0.0 |

Error handling

If an invalid floating-point number (NaN) has been created due to the calculation at output OUT, the following diagnostic event is entered in the diagnostic buffer of the F-CPU:

- "Safety program: invalid REAL number in DB" (Event ID 16#75D9)

A.2.11.4 F_DIV_R: Division of two REAL values

Function

This F-Block divides the IN1 input by the IN2 input and outputs the quotient at output OUT.

$$OUT = IN1 / IN2$$

Inputs/Outputs

| | Name | Data type | Description | Default |
|---------|------|-----------|-------------|---------|
| Inputs: | IN1 | F_REAL | Input 1 | 0.0 |
| | IN2 | F_REAL | Input 2 | 1.0 |
| Output: | OUT | F_REAL | Output | 0.0 |

Error handling

If an invalid floating-point number (NaN) has been created due to the calculation at output OUT, the following diagnostic event is entered in the diagnostic buffer of the F-CPU:

- "Safety program: invalid REAL number in DB" (Event ID 16#75D9)

See also

Behavior of F-Blocks with floating-point operations in the event of a number range overflow (Page 231)

A.2.11.5 F_ABS_R: Absolute value of a REAL value

Function

This F-Block outputs the absolute value (amount) of input IN at the output OUT.

$$OUT = | IN |$$

Inputs/Outputs

| | Name | Data type | Description | Default |
|----------------|------|-----------|-------------|---------|
| Input: | IN | F_REAL | Input | 0.0 |
| Output: | OUT | F_REAL | Output | 0.0 |

Error handling

None

A.2.11.6 F_MAX3_R: Maximum of three REAL values

Function

This F-Block compares the inputs IN1, IN2 and IN3 and outputs its maximum at output OUT. All the inputs are preset with a value of -3,402823e+38 (largest negative REAL number), so that even a maximum value can be formed from only two inputs.

$$OUT = MAX \{IN1, IN2, IN3\}$$

Inputs/Outputs

| | Name | Data type | Description | Default |
|----------------|------|-----------|-------------|---------------|
| Inputs: | IN1 | F_REAL | Input 1 | -3.402823e+38 |
| | IN2 | F_REAL | Input 2 | -3.402823e+38 |

| | Name | Data type | Description | Default |
|----------------|------|-----------|-------------|---------------|
| | IN3 | F_REAL | Input 3 | -3.402823e+38 |
| Output: | OUT | F_REAL | Output | -3.402823e+38 |

Error handling

- If one of the inputs IN1, IN2 and IN3 is an invalid floating-point number (NaN), an invalid floating-point number (NaN) is output at output OUT.
- An F-STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.11.7 F_MID3_R: Mean value of three REAL values

Function

This F-block compares inputs IN1, IN2 and IN3 and outputs the median at the OUT output.

- OUT = Median {IN1, IN2, IN3}

I/Os

| | Name | Data type | Explanation | Default |
|----------------|------|-----------|-------------|---------|
| Inputs: | IN1 | F_REAL | Input 1 | 0.0 |
| | IN2 | F_REAL | Input 2 | 0.0 |
| | IN3 | F_REAL | Input 3 | 0.0 |
| Output: | OUT | F_REAL | Output | 0.0 |

Error handling

- If one of the inputs IN1, IN2 and IN3 is an invalid floating-point number (NaN), an invalid floating-point number (NaN) is output at the OUT output.
- An F_STOP is triggered when there is an error in the safety data format in the associated instance DB. The following diagnostics event is then entered in the diagnostics buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.11.8 F_MIN3_R: Minimum of three REAL values

Function

This F-Block compares the inputs IN1, IN2 and IN3 and outputs its minimum at output OUT. All the inputs are preset with a value of 3,402823e+38 (largest positive REAL number), so that even a minimum value can be formed from only two inputs.

$$\text{OUT} = \text{MIN} \{ \text{IN1}, \text{IN2}, \text{IN3} \}$$

Inputs/Outputs

| | Name | Data type | Description | Default |
|---------|------|-----------|-------------|--------------|
| Inputs: | IN1 | F_REAL | Input 1 | 3.402823e+38 |
| | IN2 | F_REAL | Input 2 | 3.402823e+38 |
| | IN3 | F_REAL | Input 3 | 3.402823e+38 |
| Output: | OUT | F_REAL | Output | 3.402823e+38 |

Error handling

- If one of the inputs IN1, IN2 and IN3 is an invalid floating-point number (NaN), an invalid floating-point number (NaN) is output at output OUT.
- An F-STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.11.9 F_LIM_R: Asymmetrical limiter of a REAL value

Function

This F-Block checks whether input IN is within or outside the interval between MIN and MAX. If input IN lies within the interval, it is passed through to output OUT. If it lies outside of the interval it is limited to MIN or MAX.

With the F-Block you can also check the result of a floating-point operation for overflow (\pm infinity) and invalid floating-point number (NaN).

- Is $\text{IN} > \text{MAX}$ or "+ infinity", then an upper limit violation exists. MAX is output at output OUT. OUTU is set to 1 and OUTL to 0.
- Is $\text{IN} < \text{MIN}$ or "- infinity", then a lower limit violation exists. MIN is output to output OUT. OUT is set to 0 and OUTL on 1.

- If IN lies between MIN and MAX, input IN is passed through to output OUT. OUTU and OUTL are always set to 0.
- If IN is an invalid floating-point number (NaN), the fail-safe value SUBS_IN is output at output OUT. OUTU and OUTL are always set to 1.

Inputs/Outputs

| | Name | Data type | Description | Default |
|-----------------|---------|-----------|-----------------------|---------|
| Inputs: | IN | F_REAL | INPUT | 0.0 |
| | MIN | F_REAL | LOWER LIMIT | -100.0 |
| | MAX | F_REAL | UPPER LIMIT | 100.0 |
| | SUBS_IN | F_REAL | SUBSTITUTE VALUE | 0.0 |
| Outputs: | OUT | F_REAL | OUTPUT | 0.0 |
| | OUTU | F_BOOL | UPPER LIMIT VIOLATION | 0 |
| | OUTL | F_BOOL | LOWER LIMIT VIOLATION | 0 |

Error handling

- Is MIN ≥ MAX, MAX is output at output OUT. OUTU and OUTL are always set to 1.
- If one of the inputs IN, MIN, MAX or SUBS_IN is an invalid floating-point number (NaN) the fail-safe value SUBS_IN is output at output OUT. OUTU and OUTL are always set to 1.
- An F_STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.11.10 F_SQRT: Square root of a REAL value

Function

This F-Block calculates the square root of the input IN and then outputs it at the output OUT.

$$OUT = \sqrt{IN}$$

The IN input must be positive.

Inputs/outputs

| | Name | Data type | Description | Default |
|----------------|------|-----------|-------------|---------|
| Input: | IN | F_REAL | Input | 0.0 |
| Output: | OUT | F_REAL | Output | 0.0 |

Error handling

- If the calculation at output OUT yields an invalid floating-point number (NaN) or a negative value is pending at IN, NaN is output to OUT and the following diagnostic event is entered in the diagnostic buffer of the F-CPU:
 - "Safety program: invalid REAL number in DB" (Event ID 16#75D9)
- An F-STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.11.11 F_AVEX_R: Mean value of a maximum of nine REAL values

Function

This F-Block calculates the mean value from the inputs INx and outputs the result at output OUT.

$$\text{OUT} = (\text{IN1} + \text{IN2} + \dots + \text{IN8} + \text{IN9}) / 9$$

Inputs without a set validity bit VALIDINx are not included in the mean value calculation. If at least MIN inputs are valid, output VALIDOUT = 1 is set. If less than MIN inputs are valid, output VALIDOUT = 0 is set.

Inputs/Outputs

| | Name | Data type | Description | Default |
|-----------------|----------|-----------|--------------------------------|---------|
| Inputs: | IN1 | F_REAL | INPUT 1 | 0.0 |
| | ... | | ... | |
| | IN9 | F_REAL | INPUT 9 | 0.0 |
| | VALIDIN1 | F_BOOL | INPUT 1 VALID | 1 |
| | ... | | ... | |
| | VALIDIN9 | F_BOOL | INPUT 9 VALID | 1 |
| | MIN | F_INT | MINIMUM NUMBER OF VALID INPUTS | 9 |
| Outputs: | OUT | F_REAL | OUTPUT | 0.0 |
| | VALIDOUT | F_BOOL | OUTPUT VALID | 1 |

Error handling

- If an invalid floating-point number (NaN) has been created due to the calculation at output OUT, the following diagnostic event is entered in the diagnostic buffer of the F-CPU:
 - "Safety program: invalid REAL number in DB" (Event ID 16#75D9)
- An F-STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.11.12 F_SMP_AV: Sliding mean value of maximum 33 REAL values

Function

This F-Block outputs the mean value of the last N input values IN at output OUT.

$$OUT = (IN_k + IN_{k-1} + \dots + IN_{k-N+1}) / N$$

IN_k is the current input value.

The number N of input values must fulfill the condition 0 < N < 33.

Inputs/Outputs

| | Name | Data type | Description | Default |
|---------|------|-----------|----------------------------|---------|
| Inputs: | IN | F_REAL | Input | 0.0 |
| | N | F_INT | NUMBER OF INPUTS MONITORED | 1 |
| Output: | OUT | F_REAL | OUTPUT | 0.0 |

Startup characteristics

As long as N input values have not been read in after an F-Start or after an initial run, only the available input values (< N) are taken into account for averaging. Input values saved before the start are not taken into account.

Error handling

- If the condition 0 < N < 33 is not fulfilled, the current existing value at input IN is output at output OUT.
- If an invalid floating-point number (NaN) has been created due to the calculation at output OUT, the following diagnostic event is entered in the diagnostic buffer of the CPU:
 - "Safety program: invalid REAL number in DB" (Event ID 16#75D9)

- An F-STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.11.13 F_2oo3_R: Middle value of three REAL values with 2oo3 evaluation

Function

This F-block compares the three inputs IN1, IN2 and IN3 independent of the QBADx inputs and then outputs the median at the OUT output:

- $OUT = \text{Median}\{IN1, IN2, IN3\}$

If two or more INx inputs are invalid (two or more QBADx = 1), the OUT output is also invalid and the QBAD output is set to 1.

If the value of one INx input differs from the median of the three inputs IN1, IN2 and IN3 by more than the assigned tolerance DELTA, a discrepancy is detected and the DISx output is set.

So that, in the case of only one invalid INx input, its value is not output as the median at the OUT output and thus a discrepancy is detected for the invalid INx input, the fail-safe value for an invalid INx input must differ from the values typically occurring at the INx input during operation by more than the tolerance window DELTA.

I/Os

| | Name | Data type | Explanation | Default |
|-----------------|-------|-----------|---------------------------|---------|
| Inputs: | IN1 | F_REAL | Input 1 | 0.0 |
| | IN2 | F_REAL | Input 2 | 0.0 |
| | IN3 | F_REAL | Input 3 | 0.0 |
| | QBAD1 | F_BOOL | 1 = IN1 input invalid | 0 |
| | QBAD2 | F_BOOL | 1 = IN2 input invalid | 0 |
| | QBAD3 | F_BOOL | 1 = IN3 input invalid | 0 |
| | DELTA | F_REAL | Tolerance between INx | 0.0 |
| Outputs: | OUT | F_REAL | OUTPUT output | 0.0 |
| | QBAD | BOOL | 1 = OUT output is invalid | 0 |
| | DIS1 | BOOL | Discrepancy IN1 input | 0 |
| | DIS2 | BOOL | Discrepancy IN2 input | 0 |
| | DIS3 | BOOL | Discrepancy IN3 input | 0 |

Use together with F-channel driver F_CH_AI

If you interconnect the INx input of the F_2oo3_R with the V output of an F_CH_AI, you must observe the following:

1. Interconnect the QBADx input of F_2oo3_R with the QBAD output of the F_CH_AI whose V output you interconnect with the INx input of F_2oo3_R.
2. Assign the SUBS_V input of F_CH_AI with a value that differs from the values typically occurring at the INx inputs during operation by more than the tolerance window DELTA.
3. Assign the SUBS_ON input of F_CH_AI with 1.

Error handling

- If one of the inputs IN1, IN2 and IN3 is an invalid floating-point number (NaN), an invalid floating-point number (NaN) is output at the OUT output. DIS1, DIS2 and DIS3 are set to 1.
- If the DELTA input is an invalid floating-point number (NaN) or if calculations in the F-block result in invalid floating-point numbers (NaN), DIS1, DIS2 and DIS3 are set to 1.

For the case that invalid floating-point numbers (NaN) result from the calculations in the F-block, the following diagnostics event is entered in the diagnostics buffer of the F-CPU:

- "Safety program: Invalid REAL number in DB" (Event ID 16#75D9).
- An F-STOP is triggered when there is an error in the safety data format in the associated instance DB. The following diagnostics event is then entered in the diagnostics buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.11.14 F_1oo2_R: 1oo2 evaluation of inputs of data type REAL

Function

This F-Block outputs either the IN1 or IN2 input at the OUT output, depending on the QBAD1 input:

- QBAD1 = 0: OUT = IN1
- QBAD1 = 1: OUT = IN2

If both the IN1 and IN2 inputs are invalid (QBAD1 and QBAD2 = 1), the OUT output is also invalid and the QBAD output is set to 1.

If inputs IN1 and IN2 differ by more than the assigned DELTA tolerance, a discrepancy error is detected and output

- DIS1 = 1 is set if IN2 is output at the OUT output.
- DIS2 = 1 is set if IN1 is output at the OUT output.

Inputs/Outputs

| | Name | Data type | Description | Default |
|-----------------|-------|-----------|------------------------|---------|
| Inputs: | IN1 | F_REAL | Input 1 | 0.0 |
| | IN2 | F_REAL | Input 2 | 0.0 |
| | QBAD1 | F_BOOL | 1 = Input IN1 invalid | 0 |
| | QBAD2 | F_BOOL | 1 = Input IN2 invalid | 0 |
| | DELTA | F_REAL | Tolerance between INx | 0.0 |
| Outputs: | OUT | F_REAL | Output | 0.0 |
| | QBAD | F_BOOL | 1 = Output OUT invalid | 0 |
| | DIS1 | F_BOOL | Discrepancy input IN1 | 0 |
| | DIS2 | F_BOOL | Discrepancy input IN2 | 0 |

Used together with F-Channel driver F_CH_AI

If you interconnect input INx of the F_1oo2_R with output V of an F_CH_AI, you must observe the following:

- Interconnect the QBADx input of the F_1oo2_R with the QBAD output of the F_CH_AI and its output V with input INx of the F_1oo2_R.
- Configure the SUBS_V input of the F_CH_AI with a value which differs by more than the DELTA tolerance window from the values which typically occur at the INx inputs during operation.
- Configure the SUBS_ON input of the F_CH_AI with 1.

Error handling

- If one of the IN1, IN2 or DELTA inputs is an invalid floating point number (NaN) or if invalid floating-point numbers (NaNs) arise due to calculations in the F-Block, DIS1 and DIS2 are set to 1.

If invalid floating-point numbers (NaNs) arise due to calculations in the F-Block, the following diagnostic event is then entered in the diagnostic buffer of the F-CPU:

- "Safety program: Invalid REAL number in DB" (Event ID 16#75D9).
- An F-STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA).

A.2.12 Arithmetic blocks with the INT data type

Overview

| Block name | Block number | Description |
|------------|--------------|--------------------------------------|
| F_LIM_I | FB 350 | Asymmetrical limiter of an INT value |

A.2.12.1 F_LIM_I: Asymmetrical limiter of an INT value

Function

This block checks whether input IN is within or outside the interval between MIN and MAX. If input IN lies within the interval, it is passed through to output OUT. If it lies outside of the interval it is limited to MIN or MAX.

- Is $IN > MAX$, then an upper limit violation exists. MAX is output at output OUT. OUTU is set to 1 and OUTL to 0.
- If $IN < MIN$, then a lower limit violation exists. MIN is output to output OUT. OUT is set to 0 and OUTL on 1.
- If IN lies between MIN and MAX, input IN is passed through to output OUT. OUTU and OUTL are always set to 0.

Inputs/Outputs

| | Name | Data type | Description | Default |
|-----------------|------|-----------|-------------|---------|
| Inputs: | IN | F_INT | Input | 0 |
| | MIN | F_INT | MINIMUM | -32768 |
| | MAX | F_INT | MAXIMUM | 32767 |
| Outputs: | OUT | F_INT | OUTPUT | 0 |
| | OUTU | F_BOOL | UPPER LIMIT | 0 |
| | OUTL | F_BOOL | LOWER LIMIT | 0 |

Error handling

- Is $MIN \geq MAX$, MAX is output at output OUT. OUTU and OUTL are always set to 1.
- An F_STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.13 Multiplex blocks

A.2.13.1 Multiplex blocks

Overview

| Block name | Block number | Description |
|------------|--------------|---|
| F_MOV_R | FB 311 | Copy 15 values of data type REAL |
| F_MUX2_R | FB 332 | Multiplexer for 2 REAL values with BOOL selection |
| F_MUX16R | FB 334 | Multiplexer for 16 REAL values with INT selection |

A.2.13.2 F_MOV_R: Copy 15 values of data type REAL

Function

This F-Block copies the INx inputs to the OUTx outputs when input ENABLE = 1. When ENABLE = 0, the last valid values are retained at the OUTx outputs.

Output OENABLE corresponds to input ENABLE.

Inputs/outputs

| | Name | Data type | Description | Default |
|-----------------|---------|-----------|-------------------------|---------|
| Inputs: | ENABLE | F_BOOL | 1 = Enable copying | 0 |
| | IN1 | F_REAL | Input 1 | 0.0 |
| | ... | | ... | |
| | IN15 | F_REAL | Input 15 | 0.0 |
| Outputs: | OENABLE | F_BOOL | 1 = Copying is enabled | 0 |
| | OUT1 | F_REAL | Output 1 | 0.0 |
| | ... | | ... | |
| | OUT15 | F_REAL | Output 15 | 0.0 |
| | CS_USED | F_BOOL | 1 = Default values used | 0 |


Startup characteristics

Following an F-Startup, the F-Block behaves as follows:

- Following a CPU-STOP with subsequent cold restart of the F-CPU or during initial run:
When ENABLE = 0, the (assigned) default values are made available at the OUTx outputs. The CS_USED output is set to "1". CS_USED is reset to "0" as soon as ENABLE changes to "1".
When ENABLE = 1, the INx inputs are copied to the OUTx outputs. The CS_USED output is set to "0".
- Following a CPU-STOP with subsequent restart (warm restart) of the F-CPU, or following an F-STOP with subsequent positive edge at the RESTART input of the F_SHUTDOWN block:
When ENABLE = 0, the last valid values are made available at the OUTx outputs. The CS_USED output retains its default value (0).
When ENABLE = 1, the INx inputs are copied to the OUTx outputs. The CS_USED output is set to "0".

Note

Prior to initial processing of the F-Block following an F-Startup, the default values are present at outputs OUTx and CS_USED.

| |
|---|
|  WARNING |
| F-Startup Following an F-Startup, plant safety must not be compromised due to either the presence of the (assigned) default values at the OUTx outputs or the presence of the last valid values at the OUTx outputs. If necessary, evaluate the CS_USED output to determine whether the (assigned) default values or the last valid values were made available at the OUTx outputs after an F-Startup. In addition, the default value "0" of CS_USED must not be changed. If a restart (warm restart) is performed after a cold restart, CS_USED is reset to the default value (0), even if the default values are still present at the OUTx outputs. |

Error handling

An F-STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:

- "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.13.3 F_MUX2_R: Multiplexer for 2 REAL values with BOOL selection

Function

This F-Block outputs one of the IN0 or IN1 inputs, depending on selection input K, at output OUT:

- K = 0: OUT = IN0
- K = 1: OUT = IN1

Inputs/Outputs

| | Name | Data type | Description | Default |
|---------|------|-----------|-----------------|---------|
| Inputs: | K | F_BOOL | Selection input | 0 |
| | IN0 | F_REAL | Input 0 | 0.0 |
| | IN1 | F_REAL | Input 1 | 0.0 |
| Output: | OUT | F_REAL | Output | 0.0 |

Error handling

An F_STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:

- "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.13.4 F_MUX16R: Multiplexer for 16 REAL values with INT selection

Function

This block outputs one of the inputs INx, depending on selection input K, at output OUT:

- $0 \leq K \leq 15$ OUT = IN[K]

Inputs/Outputs

| | Name | Data type | Description | Default |
|---------|------|-----------|-----------------|---------|
| Inputs: | K | F_INT | Selection input | 0 |
| | IN0 | F_REAL | Input 0 | 0.0 |
| | ... | | ... | |
| | IN15 | F_REAL | Input 15 | 0.0 |
| Output: | OUT | F_REAL | Output | 0.0 |

Error handling

- If $K < 0$ or $K > 15$ 0.0 is output at output OUT.
- An F-STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)

A.2.14 F-Control blocks

Overview

| Block name | Block number | Description |
|------------|--------------|--|
| F_POLYG | FB 467 | Polyline or non-linear characteristic with 24 data points, maximum |
| F_INT_P | FB 468 | Integration function with integration and track mode |
| F_PT1_P | FB 469 | First order delay |

A.2.14.1 F_POLYG: F-Control block with non-linear characteristic

Function/mode of operation

The polygon function is used to approach any analog function by means of a specific number of intervals. These are defined by their X/Y coordinates. Within the limits of the approach, up to 24 X/Y coordinate pairs can be defined. The number of X/Y coordinate pairs must be assigned via input N.

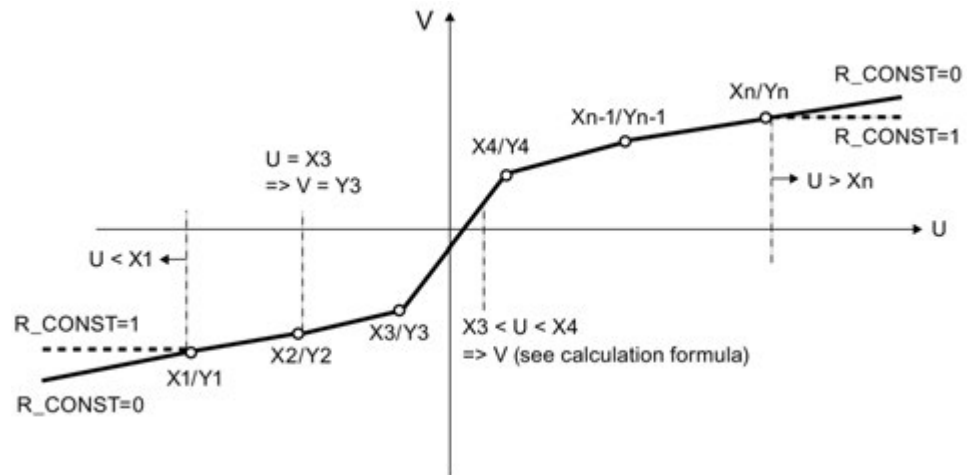
The F-Block converts input U to output V following the non-linear characteristic defined by means of the X/Y coordinate pairs, where X is the value of the analog input and Y the value of the analog output. Linear interpolation is carried out between the X_n/Y_n data points.

When $R_CONST = "0"$, extrapolation occurs outside of the end data points based on the first two and last two data points.

If $R_CONST = "1"$ and U is less than ($<$) X_1 , Y_1 is written to output V; similarly, if U is greater than ($>$) X_N , Y_N is written to output V.

In the event of an invalid parameter assignment of N ($2 > N > 24$). $V = U$ is output; the same applies for an invalid sequence of X/Y coordinate pairs ($X_n \geq X_{n+1}$ for $n = 1, 2, \dots N-1$).

The figure below provides a graphical illustration of the functionality of this F-Block.



If input value U lies between two X/Y points ($X_n < U < X_{n+1}$), V is calculated based on the following formula:

$$V = Y_n + (U - X_n) * \left(\frac{Y_{n+1} - Y_n}{X_{n+1} - X_n} \right)$$

| | |
|-------------------|------------------|
| V | Output value |
| U | Input value |
| Y_n/X_n | Data point n |
| Y_{n+1}/X_{n+1} | Data point $n+1$ |

Inputs/outputs

| | Name | Data type | Description | Default |
|-----------------|---------|-----------|---|---------|
| Inputs: | U | F_REAL | Input value | 0.0 |
| | IERR | F_BOOL | 1=input value invalid | 0 |
| | N | F_INT | Number of data points | 0 |
| | R_CONST | F_BOOL | 0=extrapolation 1=lowest/highest Y value | 0 |
| | $X1$ | F_REAL | X coordinate 1 | 0.0 |
| | $Y1$ | F_REAL | Y coordinate 1 | 0.0 |
| | : | | | |
| | $X24$ | F_REAL | X coordinate 24 | 0.0 |
| | $Y24$ | F_REAL | Y coordinate 24 | 0.0 |
| Outputs: | V | F_REAL | Output value | 0.0 |
| | QERR | F_BOOL | Output value invalid | 0 |

Error handling

The validity of input signal U is read in via input IERR. This input parameter can be connected to QBAD of the corresponding input channel driver or of a voter block.

Output QERR is set when one of the following conditions is met:

- U = NaN or one $X_n/Y_n = \text{NaN}$
NaN is assigned to output V.
- The calculation yields NaN.
NaN is assigned to output V.
- Parameter assignment error $X_n \geq X_{n+1}$
U is assigned to output V.
- Input IERR = 1

Diagnostic buffer entry

- An F-STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA).
- If an invalid REAL number is determined for V during the calculation, an entry is made in the diagnostic buffer (event ID 16#75D9).

A.2.14.2 F_INT_P: Integration function with integration and track mode

The F_INT_P F-Block works in two different modes:

- Integration mode
- TRACK mode

These two modes are described separately below.

Integration mode

Function/mode of operation

In integration mode, output V rises with a positive input signal U and falls with a negative input signal U.

This F-Block operates in integration mode by forming totals according to the trapezoidal rule for each sampling interval (Ts). The V_{internal} result achieved is located in the range $V_{\text{HL}} + \text{hyst}$ to $V_{\text{LL}} - \text{hyst}$, as shown in the figure. After being additionally limited to the range V_{LL} to V_{HL} , this value is then written to output V.

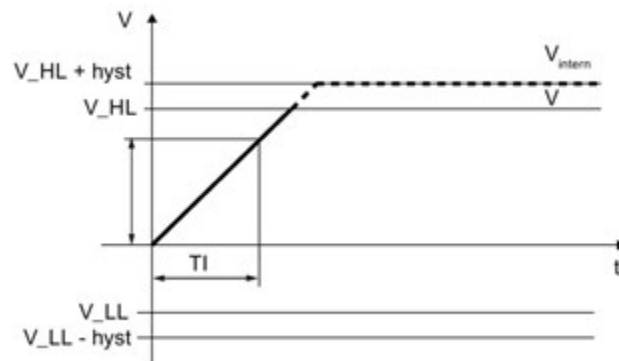


Figure A-1 Step response of F_INT_P

$$\text{hyst} = \text{HYS} / 100 * (\text{V_HL} - \text{V_LL})$$

Output value V is calculated according to the following formula:

$$V_x = V_{x-1} + U_x \cdot \frac{T_s}{T_I}$$

V_x Current internal output value

V_{x-1} Last internal output value (V_{internal})

T_s Sampling time (time elapsed between two F-Block processing cycles) in seconds

T_I Integration time in seconds

U_x Current input value

The following additional parameter assignments have an effect on output value V and its calculation:

- HOLD: When HOLD = 1, the last output value for V is held.
- RESET: When there is a positive edge at RESET, output value V is reset ($V = 0.0$).
- EN_INC and EN_DEC: Processing of the integration function also depends on the input parameters EN_INC and EN_DEC.
 - EN_INC and EN_DEC = 1
The step response at output V is rising or falling depending on U.
 - EN_INC = 0 and EN_DEC = 1:
Output value V does not rise. This means that with a positive input value at U, the last output value for V is held.
 - EN_INC = 1 and EN_DEC = 0:
Output value V does not fall. This means that with a negative input value at U, the last output value for V is held.
 - EN_INC and EN_DEC = 0:
The last output value for V is always held irrespective of input value U.

In addition to this functionality, limit value monitoring takes place:

- V_{HL} defines the upper limit for V .
If $V_{internal}$ exceeds V_{HL} , V is limited to V_{HL} ; in addition $QVHL = 1$.
- V_{LL} defines the lower limit for V .
If $V_{internal}$ falls below V_{LL} , V is limited to V_{LL} ; in addition $QVLL = 1$.

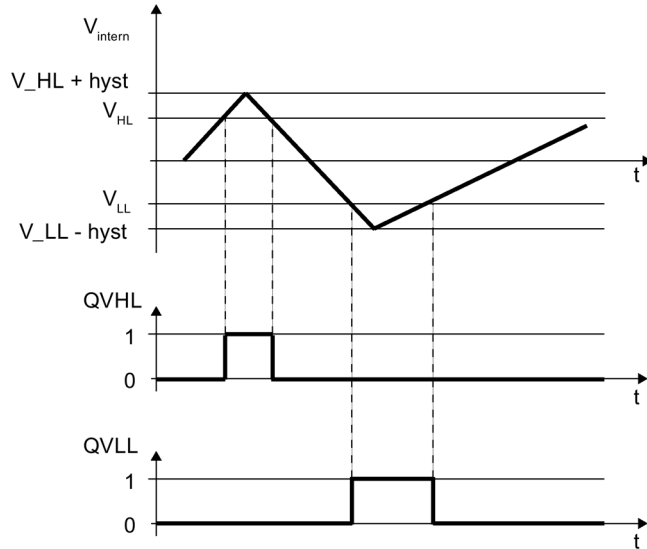


Figure A-2 Limit value monitoring of F_INT_P

Special cases:

- Hysteresis $HYS < 0$:
 HYS is set internally to 1%. $HYS = 0.0$ is permitted. In this case, $V_{internal} = V$ if V_{HL} is exceeded or V_{LL} is undershot.
- $V_{LL} > V_{HL}$:
 V_{HL} is set internally to V_{LL} . In this case, V always corresponds to V_{LL} .
- $TI \leq 0$:
 TI is set internally to Ts . Thus the times ratio assumes a value of 1 in the equation.

The validity of input signal U is read in via input $IERR$. This input parameter can be connected to $QBAD$ of the corresponding input channel driver or of a voter block.

If U , V_{HL} or V_{LL} is equal to (=) NaN , the value at output V is retained. If $HYS = NaN$, this only affects $V_{internal}$ and has no effect on V . In this case, $V_{internal} = V$. Output $QERR$ is set to 1 if NaN occurs at one of the input parameters.

Note

Denormalized values at U are processed and an error message is not output at V .

TRACK mode

In TRACK mode, input signal VTRACK is applied at output V. Thus, TRACK mode can be used to preset the integration function.

This mode is enabled via digital input TRACK = 1.

If input signal VTRACK = NaN, NaN is output at output V. The QERR output is then set to 1.

Limit value monitoring also takes place in TRACK mode:

- V_HL defines the upper limit for V.
If VTRACK exceeds V_HL, V is limited to V_HL; in addition QVHL = 1.
- V_LL defines the lower limit for V.
If VTRACK falls below V_LL, V is limited to V_LL; in addition QVLL = 1.

Special cases:

- Hysteresis HYS < 0:
HYS is set internally to 1%. HYS = 0.0 is permitted. In this case, $V_{\text{internal}} = V$ if V_HL is exceeded or V_LL is undershot. HYS has no effect on the formation of V in track mode.
- V_LL > V_HL:
V_HL is set internally to V_LL. In this case, V always corresponds to V_LL.

Inputs/outputs

| | Name | Data type | Description | Default |
|-----------------|--------|-----------|----------------------------------|---------|
| Inputs: | TI | F_TIME | Integration time | 1 s |
| | V_HL | F_REAL | High limit | 100.0 |
| | V_LL | F_REAL | Lower limit | 0.0 |
| | U | F_REAL | Input value | 0.0 |
| | HYS | F_REAL | Hysteresis in % | 1.0 |
| | VTRACK | F_REAL | Input value for track mode | 0.0 |
| | TRACK | F_BOOL | Mode: 1=track mode | 0 |
| | HOLD | F_BOOL | 1=hold integration value | 0 |
| | RESET | F_BOOL | 1=reset V | 0 |
| | EN_INC | F_BOOL | 1= rising output value permitted | 1 |
| | EN_DEC | F_BOOL | 1=falling output value permitted | 1 |
| | IERR | F_BOOL | 1=input value invalid | 0 |
| Outputs: | V | F_REAL | Output value | 0.0 |
| | QERR | F_BOOL | 1=output value invalid | 0 |
| | QVHL | F_BOOL | 1=high limit violation enabled | 0 |
| | QVLL | F_BOOL | 1=lower limit violation enabled | 0 |

Error handling

The validity of input signal U is read in via input IERR. This input parameter can be connected to QBAD of the corresponding input channel driver or of a voter block.

Output QERR is set in *integration mode* when one of the following conditions is met:

- Input signal U = NaN
- Input IERR = 1

Output QERR is set in *TRACK mode* when the following condition is met:

- VTRACK = NaN

And irrespective of the mode:

- The calculation yields NaN: Output V retains the last value.
- If NaN is present at one of the input parameters V_LL, V_HL, or HYS.

Diagnostic buffer entry

- If an invalid REAL number is determined during a calculation, an entry is made in the diagnostic buffer (event ID 16#75D9).
- An F-STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA).

A.2.14.3 F_PT1_P: First order delay

Function/mode of operation

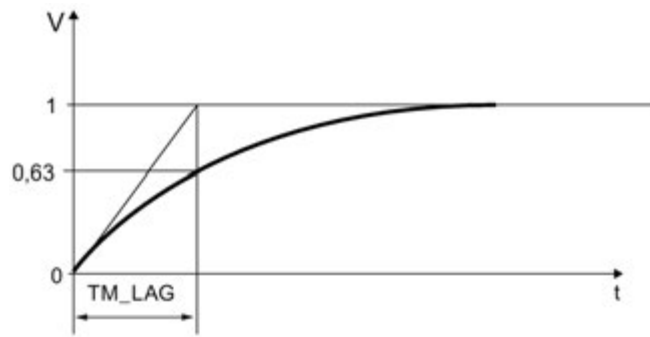
Output value V is calculated according to the following formula:

$$V_x = V_{x-1} + (U_x - V_{x-1}) \cdot \left(\frac{T_s}{\frac{T_s}{2} + TM_LAG} \right)$$

| | |
|------------------|--|
| V _x | Current output value V |
| V _{x-1} | Last output value V |
| T _s | Sampling time (time elapsed between two block processing cycles (Diff)) in seconds |
| TM_LAG | Delay time in seconds |
| U _x | Current input value U |

Input value U is output to output V with a delay corresponding to time constant TM_LAG.

The step response of an amplitude with the value U = 1.0 is reproduced in the figure below:



STOP_RES: When STOP_RES = 1, the arithmetic procedure is stopped. The last output value for V is held. During the changeover from STOP_RES 1 to 0, output V is reset to input value U.

D_OFF: When D_OFF = 1, the delay time is switched off. This means that input value U is applied at output V.

The following boundary conditions are applicable:

- $TM_LAG < T_s/2$:

TM_LAG is set to $T_s/2$. Thus the times ratio assumes a value of 1 in the equation. This means that output value V corresponds to input value U in this case.

The validity of input signal U is read in via input IERR. This input parameter can be connected to QBAD of the corresponding input channel driver or of a voter block.

Note

Denormalized values at U are processed and do not generate an error message.

If an approach to 0 occurs ($U = 0.0$), $V = 0.0$ is output when a denormalized value is reached at V ($-1.18E-38$ or $+1.18E-38$).

If U is equal to (=) NaN, the value at output V is retained. Output QERR is set to 1.

Inputs/outputs

| | Name | Data type | Description | Default |
|-----------------|----------|-----------|------------------------|---------|
| Inputs: | TM_LAG | F_TIME | Delay time | 0 s |
| | U | F_REAL | Input value | 0.0 |
| | STOP_RES | F_BOOL | Stop/reset | 0 |
| | D_OFF | F_BOOL | 1=delay switched off | 0 |
| | IERR | F_BOOL | 1=input value invalid | 0 |
| Outputs: | V | F_REAL | Output value | 0.0 |
| | QERR | F_BOOL | 1=output value invalid | 0 |

Startup characteristics

During startup input value U is applied at output V. V does not behave in accordance with PT1 behavior until a change to input value U has been made subsequently.

Error handling

Output QERR is set when one of the following conditions is met:

- Input signal U is NaN.
- The calculation yields NaN: Output V retains the last value.
- Input IERR = 1

Diagnostic buffer entry

- If an invalid REAL number is determined during the calculation, an entry is made in the diagnostic buffer (event ID 16#75D9).
- An F-STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA).

A.2.15 Additional F-Blocks

Overview

| Block name | Block number | Description |
|------------|--------------|--|
| F_DEADTM | FB 320 | Monitoring of changes in F_REAL values at the same measuring point |

A.2.15.1 F_DEADTM: Monitoring of changes in F_REAL values at the same measuring point

Function and mode of operation

This block outputs the IN value with a dead time delay at output OUT. The dead time can be configured at input DEADTM. In addition, the delta between the current IN value and the delayed IN value output at OUT is formed. This delta is output at output V_DELTA.

If the calculated delta (V_DELTA) exceeds the delta configured for input parameter DELTA by a time configured in DELAYTM, the output parameter HL (IN > OUT) or LL (IN < OUT) is activated based on the values of IN and OUT.

If 0 is configured for the DELAYTM time, output HL or LL is immediately activated as soon as the delta is exceeded.

The following boundary condition is applicable:

- If DELTA is a negative value:
The modulus is observed from DELTA.
- If DEADTM is a negative value:
DEADTM is set internally to 0.0.
- If DEADTM > 2E+8 (corresponds to approx. 6 years):
DEADTM is limited internally to 2E+8.

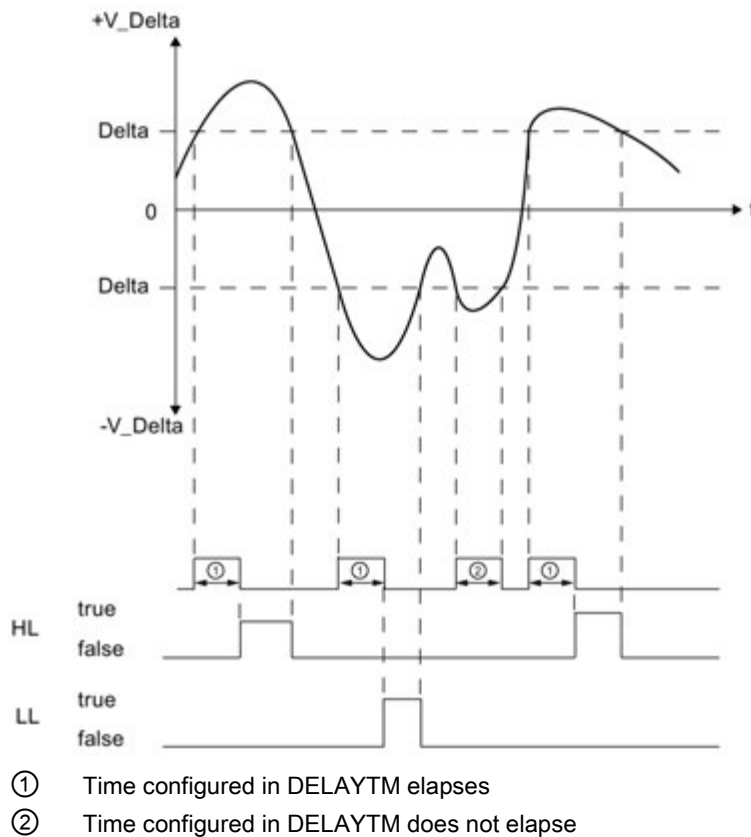


Figure A-3 Delta processing

Inputs/outputs

| | Name | Data type | Description | Default |
|---------|---------|-----------|------------------------------|---------|
| Inputs: | IN | F_REAL | Input value | 0.0 |
| | DELTA | F_REAL | Delta between IN and OUT | 0.0 |
| | DEADTM | F_REAL | Dead time | 0.0 |
| | DELAYTM | F_TIME | Delay time for HL and LL | 0 s |
| | RESTART | F_BOOL | 1=reset all values (restart) | 0 |

| | Name | Data type | Description | Default |
|-----------------|---------|-----------|----------------------------------|---------|
| Outputs: | OUT | F_REAL | Output value | 0.0 |
| | V_DELTA | F_REAL | Current delta between IN and OUT | 0.0 |
| | HL | F_BOOL | 1=delta exceeded (IN>OUT) | 0 |
| | LL | F_BOOL | 1=delta exceeded (IN<OUT) | 0 |

Startup behavior, reset

During startup or on a positive edge at input parameter RESTART, all stored values of IN are reset to the current value of IN. This IN value is output at output parameter OUT until the dead time has elapsed for the first time. Thus during the first cycle, following the events indicated above, V_DELTA is always 0 and in the following cycles until the dead time first runs out completely, V_DELTA is calculated for the time that has elapsed up to that point.

Changes to DEADTM

If changes are made to the dead time, the IN values are not output with the corresponding delay until after this time first runs out completely. During the transition time until the new dead time first runs out completely, the output values exist in relation to the old *and* new time.

Dead time tolerances

For determination of the value to be output at OUT, up to 100 different IN values can be stored within the dead time.

Values created under IN are saved and OUT and delta are processed in accordance with the OB cyclic interrupt time.

This results in the following tolerances for the dead time:

| Dead time | Max. tolerance for dead time |
|--|-----------------------------------|
| DEADTM >= 100 × OB cyclic interrupt time | DEADTM + OB cyclic interrupt time |
| DEADTM < 100 × OB cyclic interrupt time | DEADTM + (DEADTM / 100) |
| DEADTM < MAX_CYC (at F_CYC_CO) | MAX_CYC (at F_CYC_CO) |
| DEADTM < OB cyclic interrupt time | |

Error handling

The following error handling takes place for errors at input parameters DEADTM, DELTA, and IN:

- DEADTM:

When input value DEADTM = NaN, the output values of OUT and V_DELTA also become NaN, and LL and HL = 1.

- DELTA/ V_DELTA:

When input value DELTA = NaN, OUT and V_DELTA continue to be output, and LL and HL are set to 1, as comparison with DELTA is not possible.

If an invalid REAL number (NaN) is determined during the calculation of V_DELTA, the response is the same as for NaN at DELTA.

If a denormalized or infinite value is determined for V_DELTA, if it is considered a valid value. In this case, error handling does not take place.

- IN:

NaN at input IN is initially considered a normal IN value. If the dead time has elapsed and the stored NaN IN value is output to output OUT, the output values of OUT and V_DELTA become NaN, and LL and HL = 1.

Diagnostic buffer entry

- If an invalid REAL number is determined during the calculation, an entry is made in the diagnostic buffer (event ID 16#75D9).
- An F-STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA).


See also

F_CYC_CO: F-Control block "F-Cycle time monitoring" (Page 416)

A.3 S7 F Systems Lib V1_3 SP2 F-control blocks

Overview

F-Control blocks are automatically inserted and interconnected in automatically generated (F-)System charts and in automatically generated (F-)Runtime groups with ID "@F_" or "@SDW_" during compilation of the S7 program in order to create an executable safety program from the user's safety program.

| |
|--|
|  WARNING |
| Safety note - do not change automatically inserted F-Control blocks |
| Automatically inserted F-Control blocks and automatically inserted (F-)System charts and (F-)Runtime groups with ID "@F_" or "@SDW_" are visible after compilation. You must not delete them or modify them in any way (except as explicitly described). |
| Failure to adhere to this can result in errors on the next compilation. |

| Block name | Block number | Description |
|------------|--------------|--|
| F_MOVRWS | FB 312 | F-Control block |
| F_DIAG | FB 360 | F-Control block |
| F_CYC_CO | FB 395 | F-Control block "F-Cycle time monitoring" |
| F_PLK | FB 396 | F-Control block |
| F_PLK_O | FB 397 | F-Control block |
| F_TEST | FB 398 | F-Control block |
| F_TESTC | FB 399 | F-Control block |
| F_TESTM | FB 400 | F-Control block "Deactivate Safety Mode" |
| F_SHUTDN | FB 458 | F-Control block "Shutdown and F-Startup of F-Shutdown modules" |
| RTGLOGIC | FB 459 | F-Control block |
| F_PS_12 | FB 464 | F-Control block "F-Module driver" |
| F_CHG_WS | FB 477 | F-Control block |
| DB_INIT | FC 180 | F-Control block |
| DB_RES | FC 301 | F-Control block |
| F_PS_MIX | FC 302 | F-Control block |
| F_VFSTP1 | FC 307 | F-Control block |
| F_VFSTP2 | FC 308 | F-Control block |
| FORCEOFF | FC 310 | F-Control block "Deactivate F-Force" |

A.3.1 F_MOVRWS: F-Control block

Function

This F-Control block is automatically inserted and interconnected in an automatically generated system chart and in an automatically generated runtime group with ID "@SDW_" during compilation of the S7 program in order to create an executable safety program from the user's safety program.

**Safety note - do not change automatically inserted F-Control blocks**

Automatically inserted F-Control blocks and automatically inserted (F-)System charts and (F-)Runtime groups with ID "@F_" or "@SDW_" are visible after compilation. You must not delete them or modify them in any way (except as explicitly described).

Failure to adhere to this can result in errors on the next compilation.

Inputs/outputs

Non-documented inputs/outputs are initialized or interconnected automatically when the S7 program is compiled and you must not change them. Online changes affecting non-documented inputs/outputs can lead to an F-STOP. You can overcome manipulations to these inputs/outputs by recompiling the S7 program.

A.3.2 F_DIAG: F-Control block

Function

This F-Control block is automatically inserted and interconnected in an automatically generated F-System chart and in an automatically generated F-Runtime group with ID "@F_" during compilation of the S7 program in order to create an executable safety program from the user's safety program.

**Safety note - do not change automatically inserted F-Control blocks**

Automatically inserted F-Control blocks and automatically inserted (F-)System charts and (F-)Runtime groups with ID "@F_" or "@SDW_" are visible after compilation. You must not delete them or modify them in any way (except as explicitly described).

Failure to adhere to this can result in errors on the next compilation.

Inputs/outputs

Non-documented inputs/outputs are initialized or interconnected automatically when the S7 program is compiled and you must not change them. Online changes affecting non-documented inputs/outputs can lead to an F-STOP. You can overcome manipulations to these inputs/outputs by recompiling the S7 program.

A.3.3 F_CYC_CO: F-Control block "F-Cycle time monitoring"


Function

This F-Control block is automatically inserted and interconnected in an automatically generated F-System chart "@F_CycCo-OB3x" and in an automatically generated F-Runtime group with ID "@F_" during compilation of the S7 program in order to create an executable safety program from the user's safety program.

The F-CPU monitors the F-Cycle time for each cyclic interrupt OB 3x that contains F- Runtime groups. The first time you compile the S7 program, a dialog will appear and prompt you to enter a value for the maximum cycle time "MAX_CYC" that can elapse between two calls of this OB.

If you need to change the maximum F-Cycle time after the initial compilation of the S7 program, you must set the F-Cycle time at the MAX_CYC input of the F_CYC_CO-OB3x block in F-System chart @F_CycCo-OB3x.

For information about setting F-Monitoring times, refer to chapter "Run times, F-Monitoring times, and response times (Page 460)".

| |
|--|
|  WARNING |
| Safety note - do not change automatically inserted F-Control blocks |
| Automatically inserted F-Control blocks and automatically inserted (F-)System charts and (F-)Runtime groups with ID "@F_" or "@SDW_" are visible after compilation. You must not delete them or modify them in any way (except as explicitly described). |
| Failure to adhere to this can result in errors on the next compilation. |

Inputs/outputs

| | Name | Data type | Description | Default |
|--------|---------|-----------|----------------------|--|
| Input: | MAX_CYC | F_TIME | Maximum F-Cycle time | Automatically initialized with 3000 ms if no change is made in the dialog on the initial compilation |

⚠ WARNING**Default setting of the maximum MAX_CYC**

The default setting for the maximum F-Cycle time is 3000 milliseconds. Check whether this setting is appropriate for your process. Change the default setting if necessary.

Non-documented inputs/outputs are initialized or interconnected automatically when the S7 program is compiled and you must not change them. Online changes affecting non-documented inputs/outputs can lead to an F-STOP. You can overcome manipulations to these inputs/outputs by recompiling the S7 program.

Error handling

- An F-STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)
- If a safety-related error is detected, an F-STOP is triggered. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Error detected in F_CYC_CO" (event ID 16#75E1)

A.3.4 F_PLK: F-Control block**Function**

This F-Control block is automatically inserted and interconnected in an automatically generated F-System chart and in an automatically generated F-Runtime group with ID "@F_" during compilation of the S7 program in order to create an executable safety program from the user's safety program.

⚠ WARNING**Safety note - do not change automatically inserted F-Control blocks**

Automatically inserted F-Control blocks and automatically inserted (F-)System charts and (F-)Runtime groups with ID "@F_" or "@SDW_" are visible after compilation. You must not delete them or modify them in any way (except as explicitly described).

Failure to adhere to this can result in errors on the next compilation.

Inputs/outputs

Non-documented inputs/outputs are initialized or interconnected automatically when the S7 program is compiled and you must not change them. Online changes affecting non-documented inputs/outputs can lead to an F-STOP. You can overcome manipulations to these inputs/outputs by recompiling the S7 program.

Error handling

- An F-STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)
- If a safety-related error is detected, an F-STOP is triggered. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Error detected in F_PLK" (event ID 16#75E1)

A.3.5 F_PLK_O: F-Control block

Function

This F-Control block is automatically inserted and interconnected in an automatically generated F-System chart and in an automatically generated F-Runtime group with ID "@F_" during compilation of the S7 program in order to create an executable safety program from the user's safety program.

 **WARNING**

Safety note - do not change automatically inserted F-Control blocks

Automatically inserted F-Control blocks and automatically inserted (F-)System charts and (F-)Runtime groups with ID "@F_" or "@SDW_" are visible after compilation. You must not delete them or modify them in any way (except as explicitly described).

Failure to adhere to this can result in errors on the next compilation.

Inputs/outputs

Non-documented inputs/outputs are initialized or interconnected automatically when the S7 program is compiled and you must not change them. Online changes affecting non-documented inputs/outputs can lead to an F-STOP. You can overcome manipulations to these inputs/outputs by recompiling the S7 program.

Error handling

- An F-STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)
- If a safety-related error is detected, an F-STOP is triggered. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Error detected in F_PLK_O" (event ID 16#75E1)

A.3.6 F_TEST: F-Control block

Function

This F-Control block is automatically inserted and interconnected in an automatically generated F-System chart and in an automatically generated F-Runtime group with ID "@F_" during compilation of the S7 program in order to create an executable safety program from the user's safety program.



WARNING

Safety note - do not change automatically inserted F-Control blocks

Automatically inserted F-Control blocks and automatically inserted (F-)System charts and (F-)Runtime groups with ID "@F_" or "@SDW_" are visible after compilation. You must not delete them or modify them in any way (except as explicitly described).

Failure to adhere to this can result in errors on the next compilation.

Inputs/outputs

Non-documented inputs/outputs are initialized or interconnected automatically when the S7 program is compiled and you must not change them. Online changes affecting non-documented inputs/outputs can lead to an F-STOP. You can overcome manipulations to these inputs/outputs by recompiling the S7 program.


Error handling

- An F-STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)
- If a safety-related error is detected, an F-STOP is triggered. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Error detected in F_TEST" (event ID 16#75E1)

A.3.7 F_TESTC: F-Control block

Function

The F-control block is automatically inserted and interconnected in an automatically generated F-system chart and in an automatically generated F-runtime group with identifier "@F_" when the S7 program is compiled. This is done in order to generate an executable safety program from the safety program created the user.

| |
|---|
|  WARNING |
| Safety note: Do not change automatically inserted F-control blocks |
| Automatically inserted F-control blocks and automatically inserted (F-)system charts and (F-)runtime groups with the "@F_" or "@SDW_" identifier are visible after compiling. You must not delete these or make any changes to them (unless expressly described). |
| Failure to observe this may result in errors at the next compile operation. |

Note

Starting from *S7 F Systems V6.2* with *S7 F Systems Lib V1_3 SP2*, a change of the "Test cycle time" parameter in HW Config (see "CPU>Object properties>H-Parameters") and subsequent compiling of the HW configuration and safety program will cause the collective signature of your safety program to change.

I/Os

Undocumented I/Os are automatically supplied or interconnected when the S7 program is compiled and must not be changed. Online changes to undocumented I/Os can trigger an F-STOP. Remove any manipulations to these I/O by compiling the S7 program again.

Error handling

- An F-STOP is triggered when there is an error in the safety data format in the associated instance DB. The following diagnostics event is then entered in the diagnostics buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)
- If a safety-related error is detected, an F-STOP is triggered. The following diagnostics event is then entered in the diagnostics buffer of the F-CPU:
 - "Safety program: Error detected in the F_TESTC (Event ID 16#75E1)

A.3.8 F_TESTM: F-Control block "Deactivate Safety Mode"

Function

This F-Control block is automatically inserted and interconnected in an automatically generated F-System chart "@F_TestMode" and in an automatically generated F-Runtime group with ID "@F_" during compilation of the S7 program in order to create an executable safety program from the user's safety program.

At output TEST, you can evaluate whether or not safety mode is deactivated. The TEST output has the system attribute S7_m_c. It can therefore be monitored directly from an OS. This enables you to arrange to see on your display whether safety mode is deactivated.

WARNING

Safety note - do not change automatically inserted F-Control blocks

Automatically inserted F-Control blocks and automatically inserted (F-)System charts and (F-)Runtime groups with ID "@F_" or "@SDW_" are visible after compilation. You must not delete them or modify them in any way (except as explicitly described).

Failure to adhere to this can result in errors on the next compilation.

Inputs/outputs

| | Name | Data type | Description | Default |
|---------|------|-----------|-----------------------------|---------|
| Output: | TEST | BOOL | 1 = Safety mode deactivated | 0 |

Non-documented inputs/outputs are initialized or interconnected automatically when the S7 program is compiled and you must not change them. Online changes affecting non-documented inputs/outputs can lead to an F-STOP. You can overcome manipulations to these inputs/outputs by recompiling the S7 program.

Error handling

- An F-STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)
- If a safety-related error is detected, an F-STOP is triggered. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Error detected" (event ID 16#75E1)

A.3.9 F_SHUTDN: F-Control block "Control of shutdown and F-Startup of the safety program"

Function

This F-Control block is automatically inserted and interconnected in an automatically generated System chart "@F_ShutDn" and in an automatically generated F-Runtime group with ID "@F_" during compilation of the S7 program in order to create an executable safety program from the user's safety program.

This F-Control block allows you to assign the shutdown behavior and to control the shutdown and the F-Startup of the safety program.

If you have set "According to the configuration of F_SHUTDN" for the F-STOP behavior in the "Safety Program" dialog > "Shutdown Behavior" dialog, you can assign parameters for the SHUTDOWN input to specify how the safety program is to behave on an F-STOP:

- SHUTDOWN = Full: Full shutdown
- SHUTDOWN = Partial: Partial shutdown

Note

The parameter assignment for the SHUTDOWN input must not be changed during an active shutdown.

You can set input RQ_FULL = 1 to trigger a full shutdown of the safety program.

You can use a positive edge at the RESTART input to implement an F-Startup following a shutdown of the safety program (F-STOP) and elimination of the causes for the shutdown if you do not want to perform a restart (warm restart) or cold restart of the F-CPU.

Following an F-Startup, the safety program starts up automatically with the initial values. After a partial shutdown of the safety program, only the F-Shutdown groups that were in the F-STOP carry out an F-Startup. During the F-Startup, a few seconds can elapse before the initialization with the initial values is complete. During initialization, output EN_INIT = 1.

Note

After implementing an F-Startup with a positive edge at input RESTART, a user acknowledgement at input ACK_REI of the fail-safe channel drivers is required for reintegration of the F-I/O affected by the shutdown.

Output FULL_SD displays whether there is a full shutdown of the safety program. At output SD_TYP, you can read out the shutdown behavior set in the "Safety Program" dialog > "Shutdown behavior" dialog.

The SAFE_M output indicates whether the safety program is in safety mode (SAFE_M = 1) or safety mode is deactivated (SAFE_M = 0).

 **WARNING**
Safety note - do not change automatically inserted F-Control blocks

Automatically inserted F-Control blocks and automatically inserted (F-)System charts and (F-)Runtime groups with ID "@F_" or "@SDW_" are visible after compilation. You must not delete them or modify them in any way (except as explicitly described).

Failure to adhere to this can result in errors on the next compilation.

Inputs/outputs

| | Name | Data type | Description | Default |
|----------------------|-----------|-----------|---|---------|
| Inputs: | RESTART | BOOL | 1 = F-Startup after shutdown | 0 |
| | SHUTDOWN | BOOL | Shutdown behavior | Full |
| | RQ_FULL | BOOL | 1 = Trigger full shutdown | 0 |
| | ALARM_EN | BOOL | 1 = Activate messages | 1 |
| Outputs: | FULL_SD | BOOL | 1 = Full shutdown of safety program | 0 |
| | SD_TYP | BOOL | Shutdown behavior from dialog: 1 = Full shutdown | 0 |
| | EN_INIT | BOOL | 1 = Initializing safety program | 0 |
| | SAFE_M | BOOL | 1 = Safety program in safety mode | 0 |
| | F_SIG_OUT | DWORD | Collective signature of the safety program | 0 |
| | MSG_DONE | BOOL | = Output DONE of the SFB34 "ALARM_8" | 0 |
| | MSG_ERR | BOOL | = Output ERROR of the SFB34 "ALARM_8" | 0 |
| | MSG_STAT | WORD | = Output STATUS of the SFB34 "ALARM_8" | 0 |
| | MSG_ACK | WORD | = Output ACK_STATE of the SFB34 "ALARM_8" | 0 |
| | NFY_DONE | BOOL | = Output DONE of the SFB31 "NOTIFY_8P" | 0 |
| | NFY_ERR | BOOL | = Output ERROR of the SFB31 "NOTIFY_8P" | 0 |
| | NFY_STAT | WORD | = Output STATUS of the SFB31 "NOTIFY_8P" | 0 |
| Input-output: | MSG_TIME | TIME | Time for message repetition | 8h |

Non-documented inputs/outputs are initialized or interconnected automatically when the S7 program is compiled and you must not change them. Online changes affecting non-documented inputs/outputs can lead to an F-STOP. You can overcome manipulations to these inputs/outputs by recompiling the S7 program.

Message behavior

- When the safety program is shut down (an F-STOP has been triggered), the F-Control block F_SHUTDOWN issues the following messages to the OS using SFB 34 "ALARM_8" as "AS I&C message - fault with individual acknowledgement":
 - "Safety program: Partial shutdown", if a partial shutdown of one or more F-Runtime groups occurs
 - "Safety program: full shutdown", if a full shutdown of the safety program occurs
- When an F-Startup occurs after a positive edge at input RESTART, the following message is issued to the OS using SFB 31 "NOTIFY_8P" as "Operating message - no acknowledgement":
 - "F-Startup of safety program on F_SHUTDOWN"
- When safety mode is deactivated, the following message is issued to the OS using SFB 31 "NOTIFY_8P" both as "Operating message - no acknowledgement" and as "AS I&C message - fault with individual acknowledgement". The "AS I&C message" is repeated whenever time MSG_TIME expires if safety mode is still deactivated. When MSG_TIME = 0, the message is not repeated.
 - "Safety mode deactivated"

By assigning parameter 0 for input ALARM_EN, you can disable input ALARM_EN if a suitable message system is not available.

Outputs MSG_xxx and NFY_xxx

Non-fail-safe information about message behavior errors is made available for service purposes at the MSG_xxx and NFY_xxx outputs. You can read out this information on your ES/OS or, if necessary, the information can be evaluated in your standard user program. The outputs correspond to the inputs of SFB 34 "ALARM_8" or SFB 31 "NOTIFY_8P". For a description, refer to the online help for SFB 34/SFB 31 or in Manual "System Software for S7-300/400 System and Standard Functions (<http://support.automation.siemens.com/WW/view/en/1214574>)".

Error handling/diagnostic buffer entry

- If a safety-related error is detected and a full shutdown is performed (an F-STOP has been triggered), the F-Control block F_SHUTDOWN enters the following event in the diagnostic buffer of the F-CPU:
 - "Complete shutdown of the F-Program active" or "Complete shutdown of the F-Program deactivated" (event ID 16#7xDE)
- When an F-Startup occurs after a positive edge at input RESTART, the following event is entered in the diagnostic buffer of the F-CPU:
 - "Initialization of safety program start" or "Initialization of safety program end" (event ID 16#7xDF)
- When safety mode is deactivated or activated, the following event is entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Safety mode deactivated" or "Safety program: Safety mode active" (event ID 16#7xDB)

See also

F-STOP (Page 93)

F-Startup and reprogramming restart/startup protection (Page 91)

A.3.10 RTGLOGIC: F-Control block

Function

This F-Control block is automatically inserted and interconnected in an automatically generated system chart and in an automatically generated runtime group with ID "@F_" during compilation of the S7 program in order to create an executable safety program from the user's safety program.

WARNING

Safety note - do not change automatically inserted F-Control blocks

Automatically inserted F-Control blocks and automatically inserted (F-)System charts and (F-)Runtime groups with ID "@F_" or "@SDW_" are visible after compilation. You must not delete them or modify them in any way (except as explicitly described).

Failure to adhere to this can result in errors on the next compilation.

Inputs/outputs

Non-documented inputs/outputs are initialized or interconnected automatically when the S7 program is compiled and you must not change them. Online changes affecting non-documented inputs/outputs can lead to an F-STOP. You can overcome manipulations to these inputs/outputs by recompiling the S7 program.

Error handling


If you have assigned "Partial shutdown" for the shutdown behavior and a safety-related error is detected for one F-Shutdown group, the relevant F-Shutdown group is shut down (an F-STOP is triggered). The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:

- "Safety program: Shutdown of a fail-safe shutdown group" (event ID 16#7xDD)

A.3.11 F_PS_12: F-Control block "F_Module_Driver"

Function

This F-Control block is automatically inserted and interconnected in an automatically generated F-System chart "@F(x)" and in an automatically generated F-Runtime group with ID "@F_" during compilation of the S7 program in order to create an executable safety program from the user's safety program.

 **WARNING**

Safety note - do not change automatically inserted F-Control blocks

Automatically inserted F-Control blocks and automatically inserted (F-)System charts and (F-)Runtime groups with ID "@F_" or "@SDW_" are visible after compilation. You must not delete them or modify them in any way (except as explicitly described).

Failure to adhere to this can result in errors on the next compilation.

Inputs/outputs

| | Name | Data type | Description | Default |
|-----------------|-----------|-----------|-----------------------------------|---------|
| Outputs: | DIAG | DWORD | Error information | DW#16#0 |
| | PROFISAFE | F_BOOL | 1 = Communication error PROFISAFE | 0 |

Non-documented inputs/outputs are initialized or interconnected automatically when the S7 program is compiled and you must not change them. Online changes affecting non-documented inputs/outputs can lead to an F-STOP. You can overcome manipulations to these inputs/outputs by recompiling the S7 program.

Output DIAG

Non-fail-safe information about safety-related communication errors between the F-CPU and F-I/O using the PROFIsafe safety protocol are made available for service purposes at output DIAG. You can read out this information on your ES/OS or, if necessary, the information can be evaluated in your standard user program.

Structure of DIAG

| Bit no. | Assignment | Possible error causes | Remedies |
|--------------|--|---|--|
| Bit 0 | Timeout detected by F-I/O | The PROFIBUS connection between the F-CPU and F-I/O is faulty. The F-Monitoring time of the F-I/O in <i>HW Config</i> is set too low. The F-I/O is receiving invalid parameter assignment data. or | Check the PROFIBUS connection and ensure that there are no external sources of interference. Check the parameter assignment of the F-I/O in <i>HW Config</i> . If necessary, set a higher value for the F-Monitoring time. Recompile the hardware configuration, and download it to the F-CPU. Compile the S7 program again. Check the diagnostic buffer of the F-I/O. Turn the power of the F-I/O off and back on. |
| | | Internal F-I/O fault or | Replace F-I/O |
| | | Internal F-CPU fault | Replace F-CPU |
| Bit 1 | F-I/O error detected by F-I/O | See F-I/O manuals | See F-I/O manuals |
| Bit 2 | CRC error or sequence number error detected by F-I/O | See description for Bit 0 | See description for Bit 0 |
| Bit 3 | Reserve | — | — |
| Bit 4 | Timeout detected by F-System | See description for Bit 0 | See description for Bit 0 |
| Bit 5 | Sequence number error detected by F-System | See description for Bit 0 | See description for Bit 0 |
| Bit 6 | CRC error detected by F-System | See description for Bit 0 | See description for Bit 0 |
| Bit 7 | Reserve | — | — |
| Bits 8 to 31 | Reserve | — | — |

Error handling

- An F-STOP is triggered when an error occurs in the safety data format in the corresponding instance DB. The following diagnostic event is then entered in the diagnostic buffer of the F-CPU:
 - "Safety program: Error in safety data format in DB" (Event ID 16#75DA)
- The safety function requires that fail-safe values be used instead of process data for passivation of the entire F-I/O or individual channels of an F-I/O in the following cases:
 - During an F-Startup
 - When errors occur during safety-related communication (communication errors) between the F-CPU and F-I/O using the safety protocol in accordance with PROFIsafe
 - If F-I/O or channel faults are detected (e.g. wire break, short-circuit, or discrepancy error)
 - As long as you have enabled an F-I/O passivation on the F-Channel driver at input PASS_ON


One of the following diagnostic events is then entered in the diagnostic buffer of the F-CPU (except during F-Startup):

- "F-I/O input channel passivated / F-I/O input channel depassivated" (event ID 16#7xE3)
- "F-I/O output channel passivated / F-I/O output channel depassivated" (event ID 16#7xE4)
- "F-I/O passivated / F-I/O depassivated" (event ID 16#7xE5)

A.3.12 F_CHG_WS: F-Control block

Function

This F-Control block is automatically inserted and interconnected in an automatically generated system chart and in an automatically generated runtime group with ID "@SDW_" during compilation of the S7 program in order to create an executable safety program from the user's safety program.

| |
|--|
|  WARNING |
| Safety note - do not change automatically inserted F-Control blocks |
| Automatically inserted F-Control blocks and automatically inserted (F-)System charts and (F-)Runtime groups with ID "@F_" or "@SDW_" are visible after compilation. You must not delete them or modify them in any way (except as explicitly described). |
| Failure to adhere to this can result in errors on the next compilation. |

Inputs/outputs

Non-documented inputs/outputs are initialized or interconnected automatically when the S7 program is compiled and you must not change them. Online changes affecting non-documented inputs/outputs can lead to an F-STOP. You can overcome manipulations to these inputs/outputs by recompiling the S7 program.

A.3.13 DB_INIT: F-Control block

Function

This F-Control block is automatically inserted and interconnected in an automatically generated system chart and in an automatically generated runtime group with ID "@F_" during compilation of the S7 program in order to create an executable safety program from the user's safety program.

| |
|--|
|  WARNING |
|--|

| |
|--|
| Safety note - do not change automatically inserted F-Control blocks |
|--|

| |
|--|
| Automatically inserted F-Control blocks and automatically inserted (F-)System charts and (F-)Runtime groups with ID "@F_" or "@SDW_" are visible after compilation. You must not delete them or modify them in any way (except as explicitly described). |
|--|

| |
|---|
| Failure to adhere to this can result in errors on the next compilation. |
|---|


Inputs/outputs

Non-documented inputs/outputs are initialized or interconnected automatically when the S7 program is compiled and you must not change them. Online changes affecting non-documented inputs/outputs can lead to an F-STOP. You can overcome manipulations to these inputs/outputs by recompiling the S7 program.

A.3.14 DB_RES: F-Control block

Function

This F-Control block is automatically inserted and interconnected in an automatically generated system chart and in an automatically generated runtime group at the start of the runtime sequence in OB 100 with ID "@F_" during compilation of the S7 program in order to create an executable safety program from the user's safety program.

| |
|--|
|  WARNING |
| Safety note - do not change automatically inserted F-Control blocks |
| Automatically inserted F-Control blocks and automatically inserted (F-)System charts and (F-)Runtime groups with ID "@F_" or "@SDW_" are visible after compilation. You must not delete them or modify them in any way (except as explicitly described). |
| Failure to adhere to this can result in errors on the next compilation. |


Inputs/outputs

Non-documented inputs/outputs are initialized or interconnected automatically when the S7 program is compiled and you must not change them. Online changes affecting non-documented inputs/outputs can lead to an F-STOP. You can overcome manipulations to these inputs/outputs by recompiling the S7 program.

A.3.15 F_PS_MIX: F-Control block

Function

This F-Control block is automatically inserted and interconnected in an automatically generated F-System chart and in an automatically generated F-Runtime group with ID "@F_" during compilation of the S7 program in order to create an executable safety program from the user's safety program.

| |
|--|
|  WARNING |
| Safety note - do not change automatically inserted F-Control blocks |
| Automatically inserted F-Control blocks and automatically inserted (F-)System charts and (F-)Runtime groups with ID "@F_" or "@SDW_" are visible after compilation. You must not delete them or modify them in any way (except as explicitly described). |
| Failure to adhere to this can result in errors on the next compilation. |


Inputs/outputs

Non-documented inputs/outputs are initialized or interconnected automatically when the S7 program is compiled and you must not change them. Online changes affecting non-documented inputs/outputs can lead to an F-STOP. You can overcome manipulations to these inputs/outputs by recompiling the S7 program.

A.3.16 F_VFSTP1: F-Control block

Function


This F-Control block is automatically inserted in the S7 program when it is compiled in order to create an executable safety program from the user's safety program.

| |
|---|
|  WARNING |
| Safety note - do not change automatically inserted F-Control blocks Automatically inserted F-Control blocks and automatically inserted (F-)System charts and (F-)Runtime groups with ID "@F_" or "@SDW_" are visible after compilation. You must not delete them or modify them in any way (except as explicitly described). Failure to adhere to this can result in errors on the next compilation. |

A.3.17 F_VFSTP2: F-Control block

Function

This F-Control block is automatically inserted in the S7 program when it is compiled in order to create an executable safety program from the user's safety program.

| |
|---|
|  WARNING |
| Safety note - do not change automatically inserted F-Control blocks Automatically inserted F-Control blocks and automatically inserted (F-)System charts and (F-)Runtime groups with ID "@F_" or "@SDW_" are visible after compilation. You must not delete them or modify them in any way (except as explicitly described). Failure to adhere to this can result in errors on the next compilation. |

A.3.18 FORCEOFF: Deactivation of F-Force

Function

This F-Control block is automatically inserted and interconnected in an automatically generated system chart and in an automatically generated F-Runtime group with ID "@F_" during compilation of the S7 program in order to create an executable safety program from the user's safety program.



WARNING

Safety note - do not change automatically inserted F-Control blocks

Automatically inserted F-Control blocks and automatically inserted (F-)System charts and (F-)Runtime groups with ID "@F_" or "@SDW_" are visible after compilation. You must not delete them or modify them in any way (except as explicitly described).

Failure to adhere to this can result in errors on the next compilation.

Inputs/outputs

Non-documented inputs/outputs are initialized or interconnected automatically when the S7 program is compiled and you must not change them. Online changes affecting non-documented inputs/outputs can lead to an F-STOP. You can overcome manipulations to these inputs/outputs by recompiling the S7 program.

A.4 F-Library Failsafe Blocks (V1_2)

The *Failsafe Blocks* F-Library (V1_2) is the predecessor version for the *S7 F Systems Lib* F-Library V1_3.

The F-Blocks of the *Failsafe Blocks* F-Library (V1_2) are described in the online help for this F-Library.

A.5 Differences between the F-Libraries Failsafe Blocks (V1_x) and S7 F Systems Lib V1_3

The following subsections describe differences between the *Failsafe Blocks* F-Library (V1_1) and *Failsafe Blocks* F-Library (V1_2) and between the *Failsafe Blocks* F-Library (V1_2) and *S7 F Systems Lib V1_3*. Only those F-Block changes that are relevant to the user and that affect the function, including the startup behavior and error handling, and the inputs/outputs of the F-Block are described.

Even if no changes (i.e., "none") are indicated, it is possible that the signatures/initial value signatures of an F-Block have changed compared to a previous version of the F-Library, for example, due to code optimizations, changes in diagnostic buffer entries, or changes in the internal interaction of the F-Blocks.

For information about the runtimes of the F-Blocks, refer to the section entitled "Run times, F-Monitoring times, and response times (Page 460)". If required, you can find out the new memory requirements from *SIMATIC Manager*.

When you upgrade to a new version of the F-Library, take note of the F-Block changes and check whether these changes may affect the behavior of your safety program. Refer also to the section entitled "Acceptance test of safety program changes (Page 221)".

Refer to Annex 1 of the Certification Report to obtain the signatures/starting value signatures for the F-Blocks of F-Library *S7 F Systems Lib V1_3* SP1.

A.5.1 Logic blocks with the BOOL data type

| F-Blocks | <i>Failsafe Blocks</i> (V1_1) | | <i>Failsafe Blocks</i> (V1_2) | | | | Change from <i>Failsafe Blocks</i> (V1_1) to <i>Failsafe Blocks</i> (V1_2) |
|----------|-------------------------------|-------------------------|---------------------------------------|-------------------------|---|-------------------------|--|
| | | | Shipped with <i>S7 F Systems V5.2</i> | | Shipped starting with <i>S7 F Systems V5.2</i> SP1 to SP4 | | |
| | Signature | Initial value signature | Signature | Initial value signature | Signature | Initial value signature | |
| F_AND4 | 89B0 | 6837 | ← | ← | ← | ← | None |
| F_OR4 | 5DCA | 6B42 | ← | ← | ← | ← | None |
| F_XOR2 | 6D4D | 069A | ← | ← | ← | ← | None |
| F_NOT | 9CD8 | DD06 | ← | ← | ← | ← | None |
| F_2OUT3 | 34DE | D79F | ← | ← | ← | ← | None |
| F_XOUTY | 5F86 | C51D | 6A1C | C51D | ← | ← | F-STOP instead of CPU-STOP (1) |

A.5 Differences between the F-Libraries Failsafe Blocks (V1_x) and S7 F Systems Lib V1_3

1) A CPU-STOP is not triggered if a safety-related error is detected (e.g., in the safety data format). Instead, the shutdown logic shuts down either the F-Shutdown group affected by the error or the entire safety program (F-STOP).

| F-Blocks | S7 F Systems Lib V1_3 | | |
|----------|-----------------------|-------------------------|---|
| | Signature | Initial value signature | Change from <i>Failsafe Blocks (V1_2)</i> to <i>S7 F Systems Lib V1_3</i> |
| F_AND4 | 89B0 | 6837 | None |
| F_OR4 | 5DCA | 6B42 | None |
| F_XOR2 | 6D4D | 069A | None |
| F_NOT | 9CD8 | DD06 | None |
| F_2OUT3 | 34DE | D79F | None |
| F_XOUTY | 68A0 | 68BE | Default output OUTN = 1 |

A.5.2 F-Blocks for F-Communication between F-CPU's

| F-Blocks | Failsafe Blocks (V1_1) | | Failsafe Blocks (V1_2) | | | | Change from <i>Failsafe Blocks (V1_1)</i> to <i>Failsafe Blocks (V1_2)</i> |
|----------|------------------------|-------------------------|--------------------------------|-------------------------|--|-------------------------|--|
| | | | Shipped with S7 F Systems V5.2 | | Shipped starting with S7 F Systems V5.2 SP1 to SP4 | | |
| | Signature | Initial value signature | Signature | Initial value signature | Signature | Initial value signature | |
| F_SENDBO | B204 | F3D1 | E223 | F3D1 | ← | ← | F-STOP instead of CPU-STOP (1) |
| F_RCVBO | 6FFB | DCF4 | A2B9 | DCF4 | ← | ← | F-STOP instead of CPU-STOP (1) |
| F_SENDR | 3BA4 | 5B9D | 7B16 | 5B9D | ← | ← | F-STOP instead of CPU-STOP (1) |
| F_RCVR | F6F3 | 14C1 | B854 | 14C1 | ← | ← | F-STOP instead of CPU-STOP (1) |
| F_SDS_BO | — | — | — | — | — | — | — |
| F_RDS_BO | — | — | — | — | — | — | — |

A.5 Differences between the F-Libraries Failsafe Blocks (V1_x) and S7 F Systems Lib V1_3

1) A CPU-STOP is not triggered if a safety-related error is detected (e.g., in the safety data format). Instead, the shutdown logic shuts down either the F-Shutdown group affected by the error or the entire safety program (F-STOP).

| F-Blocks | S7 F Systems Lib V1_3 | | Change from <i>Failsafe Blocks (V1_2)</i> to <i>S7 F Systems Lib V1_3</i> |
|----------|-----------------------|-------------------------|---|
| | Signature | Initial value signature | |
| F_SENDBO | 8D63 | 5812 | New input EN_SMODE, For function, see block description |
| F_RCVBO | DD4B | 8360 | New output SENDMODE, For function, see block description |
| F_SENDR | 2FE2 | 678B | New input EN_SMODE, For function, see block description |
| F_RCVR | 3209 | B103 | New output SENDMODE, For function, see block description |
| F_SDS_BO | C804 | 662A | New F-Block in <i>S7 F Systems Lib V1_3</i> |
| F_RDS_BO | 4389 | EDD9 | New F-Block in <i>S7 F Systems Lib V1_3</i> |

A.5.3 F-Blocks for comparing two input values of the same type

| F-Blocks | Failsafe Blocks (V1_1) | | Failsafe Blocks (V1_2) | | | | Change from <i>Failsafe Blocks (V1_1)</i> to <i>Failsafe Blocks (V1_2)</i> |
|----------|------------------------|-------------------------|---------------------------------------|-------------------------|---|-------------------------|--|
| | Signature | Initial value signature | Shipped with <i>S7 F Systems V5.2</i> | | Shipped starting with <i>S7 F Systems V5.2 SP1 to SP4</i> | | |
| | | | Signature | Initial value signature | Signature | Initial value signature | |
| F_CMP_R | — | — | — | — | — | — | — |
| F_LIM_HL | 435E | CB3F | 5116 | 7656 | ← | ← | F-STOP instead of CPU-STOP (1) Behavior with floating-point operations (2) New input SUBS_IN, For function, see block description |
| F_LIM_LL | FB73 | CB3F | AF69 | 7656 | ← | ← | F-STOP instead of CPU-STOP (1) Behavior with floating-point operations (2) New input SUBS_IN, For function, see block description |

A.5 Differences between the F-Libraries Failsafe Blocks (V1_x) and S7 F Systems Lib V1_3

1) A CPU-STOP is not triggered if a safety-related error is detected (e.g., in the safety data format). Instead, the shutdown logic shuts down either the F-Shutdown group affected by the error or the entire safety program (F-STOP).

2) If a floating-point operation produces an overflow (\pm infinity) or a denormalized or invalid floating-point number (NaN), or if an invalid floating-point number (NaN) is already present as an address, this event no longer results in a CPU-STOP. The "Overflow (\pm infinity)", "Denormalized floating-point number", or "Invalid floating-point number (NaN)" events are:

- Either output at the output and available for further processed by the subsequent F-Blocks

or

- Signaled at special outputs. If necessary, a fail-safe value is output.

If the floating-point operation yields an invalid floating-point number (NaN) and an invalid floating-point number (NaN) does not already exist as an address, the following diagnostic event is entered in the diagnostic buffer of the F-CPU:

- "Safety program: invalid REAL number in DB" (Event ID 16#75D9)

You can use this diagnostic buffer entry to identify the F-Block with the invalid floating-point number (NaN).

Refer also to the documentation of the F-Block.

If you cannot rule out the occurrence of these events in your safety program, you must decide independently of your application whether you have to react to these events in your safety program. With F-Block F_LIM_R, you can check the result of a floating-point operation for overflow (\pm infinity) and invalid floating-point number.

| F-Blocks | <i>S7 F Systems Lib V1_3</i> | | Change from <i>Failsafe Blocks (V1_2)</i> to <i>S7 F Systems Lib V1_3</i> |
|----------|------------------------------|-------------------------|--|
| | Signature | Initial value signature | |
| F_CMP_R | 689A | 602E | New F-Block in <i>S7 F Systems Lib V1_3</i> |
| F_LIM_HL | A43A | 1E14 | If the calculations in the F-Block yield invalid floating-point numbers (NaN), the fail-safe value at input SUBS_IN is output at output QH instead of "1". Default output QHN = 1 |
| F_LIM_LL | 1451 | 1E14 | If the calculations in the F-Block yield invalid floating-point numbers (NaN), the fail-safe value at input SUBS_IN is output at output QL instead of "1". Default output QLN = 1 |

A.5.4 Voter blocks for inputs of data type REAL and BOOL

| F-Blocks | <i>Failsafe Blocks (V1_1)</i> | | <i>Failsafe Blocks (V1_2)</i> | | | | Change from <i>Failsafe Blocks (V1_1)</i> to <i>Failsafe Blocks (V1_2)</i> |
|----------|-------------------------------|-------------------------|---------------------------------------|-------------------------|---|-------------------------|--|
| | | | Shipped with <i>S7 F Systems V5.2</i> | | Shipped starting with <i>S7 F Systems V5.2 SP1 to SP4</i> | | |
| | Signature | Initial value signature | Signature | Initial value signature | Signature | Initial value signature | |
| F_2oo3DI | — | — | — | — | — | — | — |
| F_1oo2AI | — | — | — | — | — | — | — |
| F_2oo3AI | — | — | — | — | — | — | — |

| F-Blocks | <i>S7 F Systems Lib V1_3</i> | | Change from <i>Failsafe Blocks (V1_2)</i> to <i>S7 F Systems Lib V1_3</i> |
|----------|------------------------------|-------------------------|---|
| | Signature | Initial value signature | |
| F_2oo3DI | 5323 | 04A0 | New F-Block in <i>S7 F Systems Lib V1_3</i> |
| F_1oo2AI | 013D | 0CE3 | New F-Block in <i>S7 F Systems Lib V1_3</i> |
| F_2oo3AI | 4580 | CE7E | New F-Block in <i>S7 F Systems Lib V1_3</i> |

A.5.5 Blocks and F-Blocks for data conversion

| Blocks / F-Blocks | <i>Failsafe Blocks (V1_1)</i> | | <i>Failsafe Blocks (V1_2)</i> | | | | Change from <i>Failsafe Blocks (V1_1)</i> to <i>Failsafe Blocks (V1_2)</i> |
|-------------------|-------------------------------|-------------------------|---------------------------------------|-------------------------|---|-------------------------|---|
| | | | Shipped with <i>S7 F Systems V5.2</i> | | Shipped starting with <i>S7 F Systems V5.2 SP1 to SP4</i> | | |
| | Signature | Initial value signature | Signature | Initial value signature | Signature | Initial value signature | |
| F_BO_FBO | 27AB | 87DA | ← | ← | ← | ← | None |
| F_R_FR | 6ED3 | 6BCE | 4278 | 6BCE | ← | ← | Behavior with floating-point operations (2) |
| F_QUITES | 89EC | B027 | B433 | B027 | ← | ← | F-STOP instead of CPU-STOP (1) |
| F_TI_FTI | A06D | 6BCE | ← | ← | ← | ← | None |
| F_I_FI | 4871 | 87DA | ← | ← | ← | ← | None |
| F_FI_FR | — | — | — | — | 672A | 9FDE | New F-Block in <i>Failsafe Blocks (V1_2)</i> <i>S7 F Systems V5.2 SP1</i> and higher |

A.5 Differences between the F-Libraries Failsafe Blocks (V1_x) and S7 F Systems Lib V1_3

| Blocks / F-Blocks | Failsafe Blocks (V1_1) | | Failsafe Blocks (V1_2) | | | | Change from <i>Failsafe Blocks (V1_1)</i> to <i>Failsafe Blocks (V1_2)</i> |
|-------------------|------------------------|-------------------------|---------------------------------------|-------------------------|---|-------------------------|---|
| | | | Shipped with <i>S7 F Systems V5.2</i> | | Shipped starting with <i>S7 F Systems V5.2 SP1 to SP4</i> | | |
| | Signature | Initial value signature | Signature | Initial value signature | Signature | Initial value signature | |
| F_FR_FI | — | — | * | * | * | * | New F-Block in <i>Failsafe Blocks (V1_2)</i> <i>S7 F Systems V5.2</i> and higher) F-Block is not certified |
| F_CHG_R | — | — | — | — | E4CD | 5DB5 | New F-Block in <i>Failsafe Blocks (V1_2)</i> <i>S7 F Systems V5.2 SP2</i> and higher |
| F_CHG_BO | — | — | — | — | D042 | E5F2 | New F-Block in <i>Failsafe Blocks (V1_2)</i> <i>S7 F Systems V5.2 SP2</i> and higher |
| F_FBO_BO | Without | Without | Without | Without | Without | Without | None |
| F_FR_R | Without | Without | Without | Without | Without | Without | None |
| F_FI_I | Without | Without | Without | Without | Without | Without | None |
| F_FTI_TI | Without | Without | Without | Without | Without | Without | None |

1) A CPU-STOP is not triggered if a safety-related error is detected (e.g., in the safety data format). Instead, the shutdown logic shuts down either the F-Shutdown group affected by the error or the entire safety program (F-STOP).

2) If a floating-point operation produces an overflow (\pm infinity) or a denormalized or invalid floating-point number (NaN), or if an invalid floating-point number (NaN) is already present as an address, this event no longer results in a CPU-STOP. The "Overflow (\pm infinity)", "Denormalized floating-point number", or "Invalid floating-point number (NaN)" events are:

- Either output at the output and available for further processing by subsequent F-Blocks

or

- Signaled at special outputs. If necessary, a fail-safe value is output.

If the floating-point operation yields an invalid floating-point number (NaN) and an invalid floating-point number (NaN) does not already exist as an address, the following diagnostic event is entered in the diagnostic buffer of the F-CPU:

- "Safety program: invalid REAL number in DB" (Event ID 16#75D9)

You can use this diagnostic buffer entry to identify the F-Block with the invalid floating-point number (NaN).

Refer also to the documentation of the F-Block.

A.5 Differences between the F-Libraries Failsafe Blocks (V1_x) and S7 F Systems Lib V1_3

If you cannot rule out the occurrence of these events in your safety program, you must decide based on your application whether you have to react to these events in your safety program. With F-Block F_LIM_R, you can check the result of a floating-point operation for overflow (\pm infinity) and invalid floating-point number.

| Blocks / F-Blocks | S7 F Systems Lib V1_3 | | Change from <i>Failsafe Blocks (V1_2)</i> to <i>S7 F Systems Lib V1_3</i> |
|-------------------|-----------------------|-------------------------|--|
| | Signature | Initial value signature | |
| F_BO_FBO | 27AB | 87DA | None |
| F_R_FR | 4278 | 6BCE | None |
| F_QUITES | 797A | B027 | None |
| F_TL_FTI | A06D | 6BCE | None |
| F_I_FI | 4871 | 87DA | None |
| F_FI_FR | 672A | 9FDE | None |
| F_FR_FI | 2B3C | B269 | F-Block is certified New outputs OUTU and OUTL; for function, see block description |
| F_CHG_R | E4CD | 5DB5 | None |
| F_CHG_BO | D042 | E5F2 | None |
| F_FBO_BO | Without | Without | None |
| F_FR_R | Without | Without | None |
| F_FL_I | Without | Without | None |
| F_FTI_TI | Without | Without | None |

A.5.6 F-Channel drivers for F-I/O

| F-Blocks | <i>Failsafe Blocks (V1_1)</i> | | <i>Failsafe Blocks (V1_2)</i> | | | | Change from <i>Failsafe Blocks (V1_1)</i> to <i>Failsafe Blocks (V1_2)</i> |
|----------|-------------------------------|-------------------------|---------------------------------------|-------------------------|---|-------------------------|---|
| | | | Shipped with <i>S7 F Systems V5.2</i> | | Shipped starting with <i>S7 F Systems V5.2 SP1 to SP4</i> | | |
| | Signature | Initial value signature | Signature | Initial value signature | Signature | Initial value signature | |
| F_CH_BI | — | — | — | — | — | — | — |
| F_CH_BO | — | — | — | — | — | — | — |
| F_PA_AI | — | — | — | — | 9046 | 14F5 | New F-Block in <i>Failsafe Blocks (V1_2)</i> <i>S7 F Systems V5.2 SP4</i> and higher |
| F_PA_DI | — | — | — | — | BCD4 | 9564 | New F-Block in <i>Failsafe Blocks (V1_2)</i> <i>S7 F Systems V5.2 SP4</i> and higher |

A.5 Differences between the F-Libraries Failsafe Blocks (V1_x) and S7 F Systems Lib V1_3

| F-Blocks | Failsafe Blocks (V1_1) | | Failsafe Blocks (V1_2) | | | | Change from <i>Failsafe Blocks (V1_1)</i> to <i>Failsafe Blocks (V1_2)</i> |
|----------|------------------------|-------------------------|---------------------------------------|-------------------------|---|-------------------------|--|
| | | | Shipped with <i>S7 F Systems V5.2</i> | | Shipped starting with <i>S7 F Systems V5.2 SP1 to SP4</i> | | |
| | Signature | Initial value signature | Signature | Initial value signature | Signature | Initial value signature | |
| F_CH_DI | E41B | F504 | 2346 | F504 | A47F | EC21 | F-STOP instead of CPU-STOP (1) <i>S7 F Systems V5.2 SP1 and higher, new output for internal interaction; when upgrading to this version, you must perform a full download with CPU-STOP.</i> |
| F_CH_DO | 6E6A | 18CF | E0B9 | D7F0 | 92C1 | DA68 | F-STOP instead of CPU-STOP (1) New input SIM_MOD, For function, see block description <i>S7 F Systems V5.2 SP1 and higher, new output for internal interaction; when upgrading to this version, you must perform a full download with CPU-STOP.</i> |
| F_CH_AI | 296D AA4F | C540 | 8F67 | D784 | 741E | 8D4B | F-STOP instead of CPU-STOP (1) Behavior with floating-point operations (2) <i>S7 F Systems V5.2 SP1 and higher, new output for internal interaction; when upgrading to this version, you must perform a full download with CPU-STOP.</i> |

1) A CPU-STOP is not triggered if a safety-related error is detected (e.g., in the safety data format). Instead, the shutdown logic shuts down either the F-Shutdown group affected by the error or the entire safety program (F-STOP).

2) If a floating-point operation produces an overflow (\pm infinity) or a denormalized or invalid floating-point number (NaN), or if an invalid floating-point number (NaN) is already present as an address, this event no longer results in a CPU-STOP. The "Overflow (\pm infinity)", "Denormalized floating-point number", or "Invalid floating-point number (NaN)" events are:

- Either output at the output and available for further processing by subsequent F-Blocks
or
- Signaled at special outputs. If necessary, a fail-safe value is output.

A.5 Differences between the F-Libraries Failsafe Blocks (V1_x) and S7 F Systems Lib V1_3

If the floating-point operation yields an invalid floating-point number (NaN) and an invalid floating-point number (NaN) does not already exist as an address, the following diagnostic event is entered in the diagnostic buffer of the F-CPU:

- "Safety program: invalid REAL number in DB" (Event ID 16#75D9)

You can use this diagnostic buffer entry to identify the F-Block with the invalid floating-point number (NaN).

Refer also to the documentation of the F-Block.

If you cannot rule out the occurrence of these events in your safety program, you must decide based on your application whether you have to react to these events in your safety program. With F-Block F_LIM_R, you can check the result of a floating-point operation for overflow (\pm infinity) and invalid floating-point number.

| F-Blocks | S7 F Systems Lib V1_3 | | Change from <i>Failsafe Blocks (V1_2)</i> to <i>S7 F Systems Lib V1_3</i> |
|----------|-----------------------|-------------------------|---|
| | Signature | Initial value signature | |
| F_CH_BI | E888 | 5FA7 | New F-Block in <i>S7 F Systems Lib V1_3</i> |
| F_CH_BO | A8C7 | A5E4 | New F-Block in <i>S7 F Systems Lib V1_3</i> |
| F_PA_AI | 84D9 | B5A7 | Order of inputs SIM_ON and SIM_V and inputs SUBS_ON and SUBS_V reversed Output IPAR_OK F_BOOL instead of BOOL New output V_MOD, For function, see block description For behavior during F-STOP, see block description Interconnection with F_PS_12 instead of F_MPA_I |
| F_PA_DI | 2FC7 | E4F2 | Order of inputs SIM_ON and SIM_I reversed Inputs SUBS_ON and SUBS_I omitted Output IPAR_OK F_BOOL instead of BOOL New outputs QN, Q0 ... Q7 and Q_MOD, For function, see block description For behavior during F-STOP, see block description Interconnection with F_PS_12 instead of F_MPA_I |
| F_CH_DI | 3119 | EA57 | New inputs for internal interaction New outputs DISCF and DISCF_R receive discrepancy error information from output DIAG_1 and DIAG_2 of F_M_DI8 and F_M_DI24, New output Q_MOD, New outputs QMODF and QMODF_R For function, see block description In the case of redundantly configured F-I/O, a user acknowledgement is also required if the indicated errors occurred only on one F-I/O and, thus, did not trigger a fail-safe value output to the process. For behavior during F-STOP, see block description Interconnection with F_PS_12 instead of F_M_DI24 or F_M_DI8 |

A.5 Differences between the F-Libraries Failsafe Blocks (V1_x) and S7 F Systems Lib V1_3

| F-Blocks | S7 F Systems Lib V1_3 | | Change from <i>Failsafe Blocks (V1_2)</i> to <i>S7 F Systems Lib V1_3</i> |
|----------|-----------------------|----------------------------|--|
| | Signature | Initial value signature | |
| F_CH_DO | F967 | 4F58 | New inputs for internal interaction New outputs QMODF and QMODF_R For function, see block description In the case of redundantly configured F-I/O, a user acknowledgement is also required if the indicated errors occurred only on one F-I/O and, thus, did not trigger a fail-safe value output to the process. For behavior during F-STOP, see block description Interconnection with F_PS_12 instead of F_M_DO8 or F_M_DO10 |
| F_CH_AI | D846 | 3A31 | New inputs for internal interaction New input MODE taken over from F_M_AI6, New output V_MOD, New outputs QMODF and QMODF_R, New output AL_STATE For function, see block description New measuring range coding is supported, see block description In the case of redundantly configured F-I/O, a user acknowledgement is also required if the indicated errors occurred only on one F-I/O and, thus, did not trigger a fail-safe value output to the process. For behavior with floating-point operations, see block description For behavior during F-STOP, see block description Interconnection with F_PS_12 instead of F_M_AI6 |

A.5.7 F-System blocks

| Blocks / F-Blocks | Failsafe Blocks (V1_1) | | Failsafe Blocks (V1_2) | | | | Change from <i>Failsafe Blocks (V1_1)</i> to <i>Failsafe Blocks (V1_2)</i> |
|----------------------|------------------------|----------------------------|--|-------------------------------|---|----------------------------|--|
| | Signature | Initial value signature | Shipped with <i>S7 F Systems V5.2</i> | | Shipped starting with <i>S7 F Systems V5.2</i> SP1 to SP4 | | |
| | | | Signature | Initial value signature | Signature | Initial value signature | |
| F_S_BO | CC75 | 1110 | F353 | 1110 | ← | ← | None |
| F_R_BO | 3E82 D775 | B9A5 | 6CE1 | B9A5 | ← | ← | F-STOP instead of CPU-STOP (1) If no updated data are received within the F-Monitoring time, a CPU-STOP does not occur; instead, the assigned fail-safe values are output. |
| F_S_R | D897 | 1FC2 | 372C | 1FC2 | ← | ← | None |

A.5 Differences between the F-Libraries Failsafe Blocks (V1_x) and S7 F Systems Lib V1_3

| Blocks / F-Blocks | <i>Failsafe Blocks (V1_1)</i> | | <i>Failsafe Blocks (V1_2)</i> | | | | Change from <i>Failsafe Blocks (V1_1)</i> to <i>Failsafe Blocks (V1_2)</i> |
|-------------------|-------------------------------|-------------------------|---------------------------------------|-------------------------|---|-------------------------|---|
| | Signature | Initial value signature | Shipped with <i>S7 F Systems V5.2</i> | | Shipped starting with <i>S7 F Systems V5.2 SP1 to SP4</i> | | |
| | | | Signature | Initial value signature | Signature | Initial value signature | |
| F_R_R | 6C69 6F8F | 543A | 64A1 | 543A | ← | ← | F-STOP instead of CPU-STOP (1) If no updated data are received within the F-Monitoring time, a CPU-STOP does not occur; instead, the assigned fail-safe values are output. |
| F_START | 5791 | 2151 | ← | ← | ← | ← | None |
| F_PSG_M | — | — | — | — | Without | Without | New F-Block in <i>Failsafe Blocks (V1_2)</i> <i>S7 F Systems V5.2 SP1</i> and higher |

1) A CPU-STOP is not triggered if a safety-related error is detected (e.g., in the safety data format). Instead, the shutdown logic shuts down either the F-Shutdown group affected by the error or the entire safety program (F-STOP).

| Blocks / F-Blocks | <i>S7 F Systems Lib V1_3</i> | | Change from <i>Failsafe Blocks (V1_2)</i> to <i>S7 F Systems Lib V1_3</i> |
|-------------------|------------------------------|-------------------------|---|
| | Signature | Initial value signature | |
| F_S_BO | 59D5 | 1110 | None |
| F_R_BO | CC9E | E882 | None |
| F_S_R | 7394 | 1FC2 | None |
| F_R_R | AC9C | 237E | None |
| F_START | 5791 | 2151 | None |
| F_PSG_M | Without | Without | None |

A.5.8 Flip-flop blocks

| F-Blocks | <i>Failsafe Blocks (V1_1)</i> | | <i>Failsafe Blocks (V1_2)</i> | | | | Change from <i>Failsafe Blocks (V1_1)</i> to <i>Failsafe Blocks (V1_2)</i> |
|----------|-------------------------------|----------------------------|--|-------------------------------|---|----------------------------|--|
| | | | Shipped with <i>S7 F Systems V5.2</i> | | Shipped starting with <i>S7 F Systems V5.2</i> SP1 to SP4 | | |
| | Signature | Initial value signature | Signature | Initial value signature | Signature | Initial value signature | |
| F_RS_FF | 5A81 | 069A | 3A1A | 069A | ← | ← | F-STOP instead of CPU-STOP (1) |
| F_SR_FF | 7F12 | 069A | 61BC | 069A | ← | ← | F-STOP instead of CPU-STOP (1) |

1) A CPU-STOP is not triggered if a safety-related error is detected (e.g., in the safety data format). Instead, the shutdown logic shuts down either the F-Shutdown group affected by the error or the entire safety program (F-STOP).

| F-Blocks | <i>S7 F Systems Lib V1_3</i> | | Change from <i>Failsafe Blocks (V1_2)</i> to <i>S7 F Systems Lib V1_3</i> |
|----------|------------------------------|------------------------------|---|
| | Signature | Initial value signa- ture | |
| F_RS_FF | 6257 | B56D | None |
| F_SR_FF | 9EBE | B56D | None |

A.5.9 IEC pulse and counter blocks

| F-Blocks | <i>Failsafe Blocks (V1_1)</i> | | <i>Failsafe Blocks (V1_2)</i> | | | | Change from <i>Failsafe Blocks (V1_1)</i> to <i>Failsafe Blocks (V1_2)</i> |
|----------|-------------------------------|----------------------------|--|-------------------------------|---|----------------------------|--|
| | | | Shipped with <i>S7 F Systems V5.2</i> | | Shipped starting with <i>S7 F Systems V5.2</i> SP1 to SP4 | | |
| | Signature | Initial value signature | Signature | Initial value signature | Signature | Initial value signature | |
| F_CTUD | 9928 | F7D1 | EF97 | F7D1 | ← | ← | F-STOP instead of CPU-STOP (1) |
| F_TP | D608 | 7CFC | 64DD | 7CFC | ← | ← | F-STOP instead of CPU-STOP (1) |
| F_TON | DD31 | 7CFC | F8E5 | 7CFC | ← | ← | F-STOP instead of CPU-STOP (1) |
| F_TOF | F899 | 7CFC | 31A9 | 7CFC | ← | ← | F-STOP instead of CPU-STOP (1) |

A.5 Differences between the F-Libraries Failsafe Blocks (V1_x) and S7 F Systems Lib V1_3

1) A CPU-STOP is not triggered if a safety-related error is detected (e.g., in the safety data format). Instead, the shutdown logic shuts down either the F-Shutdown group affected by the error or the entire safety program (F-STOP).

| F-Blocks | S7 F Systems Lib V1_3 | | Change from <i>Failsafe Blocks (V1_2)</i> to <i>S7 F Systems Lib V1_3</i> |
|----------|-----------------------|-------------------------|---|
| | Signature | Initial value signature | |
| F_CTUD | 609B | 188C | None |
| F_TP | E671 | 22F6 | None |
| F_TON | 38DA | 22F6 | None |
| F_TOF | E45B | 22F6 | None |

A.5.10 Pulse blocks

| F-Blocks | Failsafe Blocks (V1_1) | | Failsafe Blocks (V1_2) | | | | Change from <i>Failsafe Blocks (V1_1)</i> to <i>Failsafe Blocks (V1_2)</i> |
|----------|------------------------|-------------------------|---------------------------------------|-------------------------|---|-------------------------|---|
| | Signature | Initial value signature | Shipped with <i>S7 F Systems V5.2</i> | | Shipped starting with <i>S7 F Systems V5.2 SP1 to SP4</i> | | |
| | | | Signature | Initial value signature | Signature | Initial value signature | |
| F_REPCYC | — | — | — | — | — | — | — |
| F_ROT | — | — | — | — | — | — | — |
| F_LIM_TI | 13A0 | 7CAB | 3ABB | 7CAB | ← | ← | F-STOP instead of CPU-STOP (1) |
| F_R_TRIG | 3E5E | 8F11 | BFC8 | 8F11 | ← | ← | If input CLK has a value of "1" during the first cycle after an F-Startup or an initial run, no edge is detected and output Q remains set to "0" until the next rising edge on output CLK |
| F_F_TRIG | 75E7 | 8F11 | ← | ← | ← | ← | None |

1) A CPU-STOP is not triggered if a safety-related error is detected (e.g., in the safety data format). Instead, the shutdown logic shuts down either the F-Shutdown group affected by the error or the entire safety program (F-STOP).

| F-Blocks | S7 F Systems Lib V1_3 | | Change from <i>Failsafe Blocks (V1_2)</i> to <i>S7 F Systems Lib V1_3</i> |
|----------|-----------------------|-------------------------|---|
| | Signature | Initial value signature | |
| F_REPCYC | 8F66 | 61F4 | New F-Block in <i>S7 F Systems Lib V1_3</i> |
| F_ROT | 7ECA | 73FD | New F-Block in <i>S7 F Systems Lib V1_3</i> |
| F_LIM_TI | 6E64 | 68DC | None |

A.5 Differences between the F-Libraries Failsafe Blocks (V1_x) and S7 F Systems Lib V1_3

| F-Blocks | S7 F Systems Lib V1_3 | | |
|----------|-----------------------|-------------------------|---|
| | Signature | Initial value signature | Change from <i>Failsafe Blocks (V1_2)</i> to <i>S7 F Systems Lib V1_3</i> |
| F_R_TRIG | BFC8 | 8F11 | None |
| F_F_TRIG | 75E7 | 8F11 | None |

A.5.11 Arithmetic blocks with the REAL data type

| F-Blocks | Failsafe Blocks (V1_1) | | Failsafe Blocks (V1_2) | | | | Change from <i>Failsafe Blocks (V1_1)</i> to <i>Failsafe Blocks (V1_2)</i> |
|----------|------------------------|-------------------------|--------------------------------|-------------------------|--|-------------------------|--|
| | Signature | Initial value signature | Shipped with S7 F Systems V5.2 | | Shipped starting with S7 F Systems V5.2 SP1 to SP4 | | |
| | | | Signature | Initial value signature | Signature | Initial value signature | |
| F_ADD_R | 643F | 206C | B495 | B1DF | ← | ← | Behavior with floating-point operations (2) |
| F_SUB_R | 46B5 | 206C | 5C35 | B1DF | ← | ← | Behavior with floating-point operations (2) |
| F_MUL_R | B7AC | 206C | 36DC | B1DF | ← | ← | Behavior with floating-point operations (2) |
| F_DIV_R | 9CF2 | 4A67 | D7A8 | C0B8 | ← | ← | Behavior with floating-point operations (2) |
| F_ABS_R | 7E9D | 4885 | ← | ← | ← | ← | None |
| F_MAX3_R | AEA9 | 9A67 | 78DB | 5833 | ← | ← | Behavior with floating-point operations (2) |
| F_MID3_R | 5422 | 6A94 | D596 | 6ACF | ← | ← | Behavior with floating-point operations (2) |
| F_MIN3_R | A524 | 31E1 | 551B | 2950 | ← | ← | Behavior with floating-point operations (2) |
| F_LIM_R | C92F | 0A10 | 4017 | B4BE | ← | ← | F-STOP instead of CPU-STOP (1) Behavior with floating-point operations (2) New input SUBS_IN, For function, see block description |
| F_SQRT | C412 | 895D | 593F | CDD8 | ← | ← | F-STOP instead of CPU-STOP (1) Behavior with floating-point operations (2) |

A.5 Differences between the F-Libraries Failsafe Blocks (V1_x) and S7 F Systems Lib V1_3

| F-Blocks | <i>Failsafe Blocks (V1_1)</i> | | <i>Failsafe Blocks (V1_2)</i> | | | | Change from <i>Failsafe Blocks (V1_1)</i> to <i>Failsafe Blocks (V1_2)</i> |
|----------|-------------------------------|-------------------------|---------------------------------------|-------------------------|---|-------------------------|--|
| | Signature | Initial value signature | Shipped with <i>S7 F Systems V5.2</i> | | Shipped starting with <i>S7 F Systems V5.2 SP1 to SP4</i> | | |
| | | | Signature | Initial value signature | Signature | Initial value signature | |
| F_AVEX_R | 9926 | 8CE8 | BE40 | 1CB3 | ← | ← | F-STOP instead of CPU-STOP (1) Behavior with floating-point operations (2) |
| F_SMP_AV | FB42 | 5B98 | 9D24 | 9CDF | ← | ← | F-STOP instead of CPU-STOP (1) Behavior with floating-point operations (2) |
| F_2oo3_R | — | — | FC09 | 3D43* 36CB | ← | ← | New F-Block in <i>Failsafe Blocks (V1_2)</i> <i>S7 F Systems V5.2</i> and higher *) This initial value signature is presented up to <i>S7 F Systems V5.2 SP3</i> if the block container does not contain an F-Block that was called in the F-Block type. |
| F_1oo2_R | — | — | D100 | 6717* 2ED6 | ← | ← | New F-Block in <i>Failsafe Blocks (V1_2)</i> <i>S7 F Systems V5.2</i> and higher *) This initial value signature is presented up to <i>S7 F Systems V5.2 SP3</i> if the block container does not contain an F-Block that was called in the F-Block type. |

1) A CPU-STOP is not triggered if a safety-related error is detected (e.g., in the safety data format). Instead, the shutdown logic shuts down either the F-Shutdown group affected by the error or the entire safety program (F-STOP).

2) If a floating-point operation produces an overflow (\pm infinity) or a denormalized or invalid floating-point number (NaN), or if an invalid floating-point number (NaN) is already present as an address, this event no longer results in a CPU-STOP. The "Overflow (\pm infinity)", "Denormalized floating-point number", or "Invalid floating-point number (NaN)" events are:

- Either output at the output and available for further processing by subsequent F-Blocks
or
- Signaled at special outputs. If necessary, a fail-safe value is output.

If the floating-point operation yields an invalid floating-point number (NaN) and an invalid floating-point number (NaN) does not already exist as an address, the following diagnostic event is entered in the diagnostic buffer of the F-CPU:

- "Safety program: invalid REAL number in DB" (Event ID 16#75D9)

A.5 Differences between the F-Libraries Failsafe Blocks (V1_x) and S7 F Systems Lib V1_3

You can use this diagnostic buffer entry to identify the F-Block with the invalid floating-point number (NaN).

Refer also to the documentation of the F-Block.

If you cannot rule out the occurrence of these events in your safety program, you must decide based on your application whether you have to react to these events in your safety program. With F-Block F_LIM_R, you can check the result of a floating-point operation for overflow (\pm infinity) and invalid floating-point number.

| F-Blocks | S7 F Systems Lib V1_3 | | |
|----------|-----------------------|-------------------------|--|
| | Signature | Initial value signature | Change from <i>Failsafe Blocks (V1_2)</i> to <i>S7 F Systems Lib V1_3</i> |
| F_ADD_R | DFBF | B1DF | None |
| F_SUB_R | E217 | B1DF | None |
| F_MUL_R | AA0F | B1DF | None |
| F_DIV_R | 43F6 | C0B8 | None |
| F_ABS_R | 7E9D | 4885 | None |
| F_MAX3_R | C14F | F93F | F-STOP when an error occurs in the safety data format in the instance DB |
| F_MID3_R | EC2C | EA98 | F-STOP when an error occurs in the safety data format in the instance DB |
| F_MIN3_R | D0D7 | E12A | F-STOP when an error occurs in the safety data format in the instance DB |
| F_LIM_R | B3D0 | 3957 | None |
| F_SQRT | E621 | 6B0F | None |
| F_AVEX_R | E57D | 947D | None |
| F_SMP_AV | 5659 | EEDA | None |
| F_2oo3_R | AB9F | 112C | In <i>S7 F Systems Lib V1_3</i> and higher, the F-Block is not an F-Block type Data type output DELTA is F_REAL |
| F_1oo2_R | DA53 | AA5A | In <i>S7 F Systems Lib V1_3</i> and higher, the F-Block is not an F-Block type Data type output DELTA is F_REAL |

A.5.12 Arithmetic blocks with the INT data type

| F-Block | <i>Failsafe Blocks (V1_1)</i> | | <i>Failsafe Blocks (V1_2)</i> | | | | Change from <i>Failsafe Blocks (V1_1)</i> to <i>Failsafe Blocks (V1_2)</i> |
|---------|-------------------------------|----------------------------|--|-------------------------------|---|----------------------------|--|
| | | | Shipped with <i>S7 F Systems V5.2</i> | | Shipped starting with <i>S7 F Systems V5.2</i> SP1 to SP4 | | |
| | Signature | Initial value signature | Signature | Initial value signature | Signature | Initial value signature | |
| F_LIM_I | 5219 | F4F9 | 0B0C | F4F9 | ← | ← | F-STOP instead of CPU-STOP (1) |

1) A CPU-STOP is not triggered if a safety-related error is detected (e.g., in the safety data format). Instead, the shutdown logic shuts down either the F-Shutdown group affected by the error or the entire safety program (F-STOP).

| F-Block | <i>S7 F Systems Lib V1_3</i> | | Change from <i>Failsafe Blocks (V1_2)</i> to <i>S7 F Systems Lib V1_3</i> |
|---------|------------------------------|------------------------------|---|
| | Signature | Initial value signa- ture | |
| F_LIM_I | 4845 | 4D9B | None |

A.5.13 Multiplex blocks

| F-Blocks | <i>Failsafe Blocks (V1_1)</i> | | <i>Failsafe Blocks (V1_2)</i> | | | | Change from <i>Failsafe Blocks (V1_1)</i> to <i>Failsafe Blocks (V1_2)</i> |
|----------|-------------------------------|----------------------------|--|-------------------------------|---|----------------------------|--|
| | | | Shipped with <i>S7 F Systems V5.2</i> | | Shipped starting with <i>S7 F Systems V5.2</i> SP1 to SP4 | | |
| | Signature | Initial value signature | Signature | Initial value signature | Signature | Initial value signature | |
| F_MOV_R | — | — | — | — | — | — | — |
| F_MUX2_R | 5911 | 5B43 | 7DE0 | 5B43 | ← | ← | F-STOP instead of CPU-STOP (1) |
| F_MUX16R | — | — | — | — | — | — | — |

A.5 Differences between the F-Libraries Failsafe Blocks (V1_x) and S7 F Systems Lib V1_3

1) A CPU-STOP is not triggered if a safety-related error is detected (e.g., in the safety data format). Instead, the shutdown logic shuts down either the F-Shutdown group affected by the error or the entire safety program (F-STOP).

| F-Blocks | S7 F Systems Lib V1_3 | | |
|----------|-----------------------|-------------------------|---|
| | Signature | Initial value signature | Change from <i>Failsafe Blocks (V1_2)</i> to <i>S7 F Systems Lib V1_3</i> |
| F_MOV_R | 652F | C02B | New F-Block in <i>S7 F Systems Lib V1_3</i> |
| F_MUX2_R | BFE3 | 9CB1 | None |
| F_MUX16R | AF74 | EEFE | New F-Block in <i>S7 F Systems Lib V1_3</i> |

A.5.14 F-Control blocks

| Blocks / F-Blocks | <i>Failsafe Blocks (V1_1)</i> | | <i>Failsafe Blocks (V1_2)</i> | | | | Change from <i>Failsafe Blocks (V1_1)</i> to <i>Failsafe Blocks (V1_2)</i> |
|-------------------|-------------------------------|-------------------------|---------------------------------------|-------------------------|---|-------------------------|---|
| | | | Shipped with <i>S7 F Systems V5.2</i> | | Shipped starting with <i>S7 F Systems V5.2 SP1 to SP4</i> | | |
| | Signature | Initial value signature | Signature | Initial value signature | Signature | Initial value signature | |
| F_MOVRWS | — | — | — | — | — | — | — |
| F_MPA_I | — | — | — | — | F0D1 | 381B | New F-Block in <i>Failsafe Blocks (V1_2)</i> <i>S7 F Systems V5.2 SP4</i> and higher |
| F_DIAG | — | — | — | — | — | — | — |
| F_M_DI8 | 4996 | 640D | 8FA4 | 9D22 | 5078 | 94DC | F-STOP instead of CPU-STOP (1) <i>S7 F Systems V5.2</i> and higher, new outputs PROFISAFE1 and PROFISAFE2, For function, see block description <i>S7 F Systems V5.2 SP1</i> and higher, change so that <i>S7-PLCSIM</i> can be used even without F-Simulation blocks |

A.5 Differences between the F-Libraries Failsafe Blocks (V1_x) and S7 F Systems Lib V1_3

| Blocks / F-Blocks | Failsafe Blocks (V1_1) | | Failsafe Blocks (V1_2) | | | | Change from <i>Failsafe Blocks (V1_1)</i> to <i>Failsafe Blocks (V1_2)</i> |
|-------------------|------------------------|-------------------------|---------------------------------------|-------------------------|---|-------------------------|---|
| | | | Shipped with <i>S7 F Systems V5.2</i> | | Shipped starting with <i>S7 F Systems V5.2 SP1 to SP4</i> | | |
| | Signature | Initial value signature | Signature | Initial value signature | Signature | Initial value signature | |
| F_M_DI24 | 7DA1 | 0D91 | EB16 | 1FE2 | F887 | 2EAC | F-STOP instead of CPU-STOP (1) <i>S7 F Systems V5.2</i> and higher, new outputs PROFISAFE1 and PROFISAFE2, For function, see block description <i>S7 F Systems V5.2 SP1</i> and higher, change so that <i>S7-PLCSIM</i> can be used even without F-Simulation blocks |
| F_M_DO10 | A89E | EE4E | 22E8 | EB44 | 6CA7 | 4A6E | F-STOP instead of CPU-STOP (1) <i>S7 F Systems V5.2</i> and higher, new outputs PROFISAFE1 and PROFISAFE2, For function, see block description <i>S7 F Systems V5.2 SP1</i> and higher, change so that <i>S7-PLCSIM</i> can be used even without F-Simulation blocks |
| F_M_AI6 | 3CC4 | 75CE | AF64 | EC0D | 1E41 | D818 | F-STOP instead of CPU-STOP (1) <i>S7 F Systems V5.2</i> and higher, new outputs PROFISAFE1 and PROFISAFE2, For function, see block description <i>S7 F Systems V5.2 SP1</i> and higher, change so that <i>S7-PLCSIM</i> can be used even without F-Simulation blocks |
| F_M_DO8 | — | — | 7337 | 3B1F | 86EF | BD24 | New F-Block in <i>Failsafe Blocks (V1_2)</i> <i>S7 F Systems V5.2</i> and higher <i>S7 F Systems V5.2 SP1</i> and higher, change so that <i>S7-PLCSIM</i> can be used even without F-Simulation blocks |
| F_CYC_CO | 3263 | CB5D | E895 | 6769 | ← | ← | F-STOP instead of CPU-STOP (1) |

A.5 Differences between the F-Libraries Failsafe Blocks (V1_x) and S7 F Systems Lib V1_3

| Blocks / F-Blocks | Failsafe Blocks (V1_1) | | Failsafe Blocks (V1_2) | | | | Change from <i>Failsafe Blocks (V1_1)</i> to <i>Failsafe Blocks (V1_2)</i> |
|----------------------|------------------------|----------------------------|--|----------------------------|---|----------------------------|---|
| | | | Shipped with <i>S7 F Systems V5.2</i> | | Shipped starting with <i>S7 F Systems V5.2</i> SP1 to SP4 | | |
| | Signature | Initial value signature | Signature | Initial value signature | Signature | Initial value signature | |
| F_PLK | E5B4 | D2F9 | A234 | 5FA0 | ← | ← | F-STOP instead of CPU-STOP (1) |
| F_PLK_O | 53BE | 3E43 | D690 | 834C | ← | ← | F-STOP instead of CPU-STOP (1) |
| F_TEST | D774 | A04B | 5B6D | 38AF | ← | ← | F-STOP instead of CPU-STOP (1) |
| F_TESTC | E7E8 | 711C | 5A93 | D8AA | ← | ← | F-STOP instead of CPU-STOP (1) |
| F_TESTM | 2983 | BED2 | ← | ← | ← | ← | None |
| F_SHUTDN | — | — | Without | Without | Without | Without | New F-Block in <i>Failsafe Blocks (V1_2)</i> <i>S7 F Systems V5.2</i> and higher |
| RTGLOGIC | — | — | Without | Without | Without | Without | New F-Block in <i>Failsafe Blocks (V1_2)</i> <i>S7 F Systems V5.2</i> and higher |
| F_PS_12 | — | — | — | — | — | — | — |
| F_CHG_WS | — | — | — | — | Without | Without | New F-Block in <i>Failsafe Blocks (V1_2)</i> <i>S7 F Systems V5.2</i> and higher |
| DB_INIT | — | — | Without | Without | Without | Without | New F-Block in <i>Failsafe Blocks (V1_2)</i> <i>S7 F Systems V5.2</i> SP2 or later |
| FAIL_MSG | — | — | Without | Without | Without | Without | New F-Block in <i>Failsafe Blocks (V1_2)</i> <i>S7 F Systems V5.2</i> and higher |
| DB_RES | Without | Without | Without | Without | Without | Without | None |
| F_PS_MIX | — | — | — | — | — | — | — |
| F_VFSTP1 | — | — | — | — | — | — | — |
| F_VFSTP2 | — | — | — | — | — | — | — |

A.5 Differences between the F-Libraries Failsafe Blocks (V1_x) and S7 F Systems Lib V1_3

1) A CPU-STOP is not triggered if a safety-related error is detected (e.g., in the safety data format). Instead, the shutdown logic shuts down either the F-Shutdown group affected by the error or the entire safety program (F-STOP).

| Blocks / F-Blocks | S7 F Systems Lib V1_3 | | Change from <i>Failsafe Blocks (V1_2)</i> to <i>S7 F Systems Lib V1_3</i> |
|-------------------|-----------------------|-------------------------|--|
| | Signature | Initial value signature | |
| F_MOVRWS | Without | Without | New block in <i>S7 F Systems Lib V1_3</i> |
| F_MPA_I | — | — | F-Block omitted as of <i>S7 F Systems Lib V1_3</i> and is replaced with F_PS_12 |
| F_DIAG | 40FC | DDF4 | New F-Block in <i>S7 F Systems Lib V1_3</i> |
| F_M_DI8 | — | — | F-Block omitted as of <i>S7 F Systems Lib V1_3</i> and is replaced with F_PS_12. In redundantly configured F-I/O, this F-Block is replaced with two instances of F_PS_12. Inputs DISC_ON, DISCTIME and RED in <i>S7 F Systems Lib V1_3</i> and higher on F_CH_DI Discrepancy error information from output DIAG_1 and DIAG_2 in <i>S7 F Systems Lib V1_3</i> and higher at F_CH_DI output DISCF or DISCF_R Output DIAG_1/2 and PROFISAFE1/2 are on F_PS_12 output DIAG and PROFISAFE, respectively. |
| F_M_DI24 | — | — | F-Block omitted as of <i>S7 F Systems Lib V1_3</i> and is replaced with F_PS_12. In redundantly configured F-I/O, this F-Block is replaced with two instances of F_PS_12. Inputs DISC_ON, DISCTIME and RED in <i>S7 F Systems Lib V1_3</i> and higher on F_CH_DI Discrepancy error information from output DIAG_1 and DIAG_2 in <i>S7 F Systems Lib V1_3</i> and higher at F_CH_DI output DISCF or DISCF_R Output DIAG_1/2 and PROFISAFE1/2 are on F_PS_12 output DIAG and PROFISAFE, respectively. |
| F_M_DO10 | — | — | F-Block omitted as of <i>S7 F Systems Lib V1_3</i> and is replaced with F_PS_12. In redundantly configured F-I/O, this F-Block is replaced with two instances of F_PS_12. Input RED in <i>S7 F Systems Lib V1_3</i> and higher on F_CH_DO |
| F_M_AI6 | — | — | F-Block omitted as of <i>S7 F Systems Lib V1_3</i> and is replaced with F_PS_12. In redundantly configured F-I/O, this F-Block is replaced with two instances of F_PS_12. Inputs MODE_xx in <i>S7 F Systems Lib V1_3</i> and higher on F_CH_AI as input MODE Input RED in <i>S7 F Systems Lib V1_3</i> and higher on F_CH_AI |
| F_M_DO8 | — | — | F-Block omitted as of <i>S7 F Systems Lib V1_3</i> and is replaced with F_PS_12. In redundantly configured F-I/O, this F-Block is replaced with two instances of F_PS_12. Input RED in <i>S7 F Systems Lib V1_3</i> and higher on F_CH_DO |
| F_CYC_CO | 701D | 424E | None |
| F_PLK | CD05 | A65D | None |
| F_PLK_O | 45F2 | 7B78 | None |

A.5 Differences between the F-Libraries Failsafe Blocks (V1_x) and S7 F Systems Lib V1_3

| Blocks / F-Blocks | S7 F Systems Lib V1_3 | | Change from <i>Failsafe Blocks (V1_2)</i> to <i>S7 F Systems Lib V1_3</i> |
|----------------------|-----------------------|-------------------------|---|
| | Signature | Initial value signature | |
| F_TEST | EC5F | EB03 | None |
| F_TESTC | 680A | 38BA | None |
| F_TESTM | 8B5A | 9A74 | Message behavior is taken over from F_SHUTDOWN in <i>S7 F Systems Lib V1_3</i> and higher |
| F_SHUTDOWN | Without | Without | New output SD_TYP, New input/output MSG_TIME, For function, see block description Parameter assignment at input SHUTDOWN is only relevant if "Based on F_SHUTDOWN parameter assignment" is specified for the F-STOP behavior in the "Safety Program" dialog > "Shutdown Behavior" dialog. See block description |
| RTGLOGIC | Without | Without | Name changed from RTG_LOGIC to RTGLOGIC |
| F_PS_12 | A56A | B87A | New F-Block in <i>S7 F Systems Lib V1_3</i> |
| F_CHG_WS | Without | Without | None |
| DB_INIT | Without | Without | None |
| FAIL_MSG | — | — | Block omitted as of <i>S7 F Systems Lib V1_3</i> |
| DB_RES | Without | Without | None |
| F_PS_MIX | AD87 | Without | New F-Block in <i>S7 F Systems Lib V1_3</i> |
| F_VFSTP1 | Without | Without | New block in <i>S7 F Systems Lib V1_3</i> |
| F_VFSTP2 | Without | Without | New block in <i>S7 F Systems Lib V1_3</i> |

A.6 Differences between the S7 F Systems Lib F-libraries

A.6.1 Differences between the S7 F Systems Lib V1_3 SP1 and SP2 F-libraries

Overview

The following description points out the differences between the *S7 F Systems Lib V1_3 SP1* and *V1_3 SP2* F-libraries

Only the user-relevant changes of the F-blocks that affect the function, including start-up behavior and error handling, and the I/Os of the F-block are described.

Even if "None" is indicated for the change, the signatures/initial value signatures of an F-block may have changed compared to a previous version of the F-library, e.g. due to code optimizations, changes in diagnostics buffer entries or changes in the internal interaction of the F-blocks.

For information on the runtimes of the F-blocks, refer to section "Run times, F-Monitoring times, and response times (Page 460)",

You can obtain changes in the memory requirement, if needed, from SIMATIC Manager.

Note the changes of the F-blocks following a migration to a new version of the F-library and check whether the described changes have any effects on the behavior of your safety program. Note also section "Acceptance test of safety program changes (Page 221)".

Note

Starting from S7 F/FH Systems V6.2 with S7 F Systems Lib V1_3 SP2, a change of the "Test cycle time" parameter (CPU>Object properties>H-Parameters) in HW Config and subsequent compiling of the HW configuration and safety program will cause the collective signature of your safety program to change.

You can obtain the signatures/initial value signatures for the F-blocks of the S7 F Systems Lib V1_3 SP2 in Annex 1 of the Certificate Report.

| F-blocks | <i>S7 F Systems Lib V1_3 SP2</i> | | Delta download-capable | Change from <i>S7 F Systems Lib V1_3 SP1</i> to <i>SP2</i> |
|----------|----------------------------------|-------------------------|------------------------|---|
| | Signature | Initial value signature | | |
| F_2oo3AI | Annex 1 | Annex 1 | — | MODE input added Changed MED_MAX/MED_MIN Changed OUT_AVG behavior |
| F_CH_AI | Annex 1 | Annex 1 | — | Discrepancy analysis added |
| F_CH_RI | Annex 1 | Annex 1 | — | New block in S7 F Systems Lib V1_3 SP2 |
| F_PLK | Annex 1 | Annex 1 | — | None |
| F_RCVBO | Annex 1 | Annex 1 | — | New COMMVER_USED input |
| F_RDS_BO | Annex 1 | Annex 1 | — | New COMMVER_USED input |
| F_SDS_BO | Annex 1 | Annex 1 | Yes | None |

| F-blocks | S7 F Systems Lib V1_3 SP2 | | Delta download-capable | Change from S7 F Systems Lib V1_3 SP1 to SP2 |
|----------|---------------------------|-------------------------|------------------------|---|
| | Signature | Initial value signature | | |
| F_SENDR | Annex 1 | Annex 1 | Yes | None |
| F_SWC_BO | Annex 1 | Annex 1 | Yes | Behavior when interconnected with SWC_MOS and SWC_QOS |
| F_SWC_CB | Annex 1 | Annex 1 | — | New F-block in S7 F Systems Lib V1_3 SP2 |
| F_SWC_CR | Annex 1 | Annex 1 | — | New F-block in S7 F Systems Lib V1_3 SP2 |
| F_SWC_P | 7add* | 5a86* | Yes | ADR_OPESA output made visible |
| F_TESTC | Annex 1 | Annex 1 | — | None |
| F_XoutY | Annex 1 | Annex 1 | — | OUT_AE output added |
| F_RCVR | Annex 1 | Annex 1 | — | New COMMVER_USED input |
| F_SENDBO | Annex 1 | Annex 1 | Yes | None |
| SWC_CHG | Without | Without | — | New F-block in S7 F Systems Lib V1_3 SP2 |
| SWC_MOS | Without | Without | — | ADR_SWC input made visible |
| SWC_QOS | Without | Without | — | New F-block in S7 F Systems Lib V1_3 SP2 |

* The change that was made was not signature-relevant; therefore, the signatures have not changed.

A.6.2 Differences between the F-Library S7 F Systems Lib V1_3 and V1_3 SP1

The following subsections describe the differences between the F-Library *S7 F Systems Lib V1_3* and *V1_3 SP1*. Only those F-Block changes that are relevant to the user and that affect the function, including the startup behavior and error handling, and the inputs/outputs of the F-Block are described.

Even if no changes (i.e., "none") are indicated, it is possible that the signatures/initial value signatures of an F-Block have changed compared to a previous version of the F-Library, for example, due to code optimizations, changes in diagnostic buffer entries, or changes in the internal interaction of the F-Blocks.

For information about the runtimes of the F-Blocks, refer to the section entitled "Run times, F-Monitoring times, and response times (Page 460)". If required, you can find out the new memory requirements from *SIMATIC Manager*.

When you upgrade to a new version of the F-Library, take note of the F-Block changes and check whether these changes may affect the behavior of your safety program. Refer also to the section entitled "Acceptance test of safety program changes (Page 221)".

A.6 Differences between the S7 F Systems Lib F-libraries

Refer to Annex 1 of the Certification Report to obtain the signatures/starting value signatures for the F-Blocks of F-Library *S7 F Systems Lib V1_3 SP1*.

| F-Blocks | <i>S7 F Systems Lib V1_3 SP1</i> | | Delta download-capable | Change from <i>S7 F Systems Lib V1_3 to V1_3 SP1</i> |
|----------|----------------------------------|-------------------------|--|--|
| | Signature | Initial value signature | | |
| F_FR_FDI | Annex 1 | Annex 1 | — | New F-Block in <i>S7 F Systems Lib V1_3 SP1</i> |
| F_FDI_FR | Annex 1 | Annex 1 | — | New F-Block in <i>S7 F Systems Lib V1_3 SP1</i> |
| F_QUITES | Annex 1 | Annex 1 | Yes | None |
| F_CHG_BO | D042 * | E5F2 * | Yes | None |
| F_CHG_R | E4CD * | 5DB5 * | Yes | None |
| F_CH_BI | Annex 1 | Annex 1 | Yes | IPAR_EN and IPAR_OK visible |
| F_CH_BO | A8C7 * | A5E4 * | Yes | IPAR_EN and IPAR_OK visible |
| F_PA_AI | Annex 1 | Annex 1 | Yes | IPAR_EN and IPAR_OK visible and update of V_MOD |
| F_PA_DI | 2FC7 * | E4F2 * | Yes | IPAR_EN and IPAR_OK visible |
| F_CH_DO | Annex 1 | Annex 1 | With this change, the F-Channel driver F_CH_DO in <i>S7 F Systems V6.1</i> or earlier can no longer be compiled. | The output of ACK_REQ has been delayed. |
| F_CH_AI | Annex 1 | Annex 1 | Yes | IPAR_EN and IPAR_OK visible and update of V_MOD and AL_STATE |
| F_CH_II | Annex 1 | Annex 1 | — | New F-Block in <i>S7 F Systems Lib V1_3 SP1</i> |
| F_CH_IO | Annex 1 | Annex 1 | — | New F-Block in <i>S7 F Systems Lib V1_3 SP1</i> |
| F_CH_DII | Annex 1 | Annex 1 | — | New F-Block in <i>S7 F Systems Lib V1_3 SP1</i> |
| F_CH_DIO | Annex 1 | Annex 1 | — | New F-Block in <i>S7 F Systems Lib V1_3 SP1</i> |
| F_SQRT | Annex 1 | Annex 1 | Yes | None |
| F_POLYG | Annex 1 | Annex 1 | — | New F-Block in <i>S7 F Systems Lib V1_3 SP1</i> |
| F_INT_P | Annex 1 | Annex 1 | — | New F-Block in <i>S7 F Systems Lib V1_3 SP1</i> |
| F_PT1_P | Annex 1 | Annex 1 | — | New F-Block in <i>S7 F Systems Lib V1_3 SP1</i> |
| F_SWC_P | Annex 1 | Annex 1 | — | New F-Block in <i>S7 F Systems Lib V1_3 SP1</i> |
| F_SWC_BO | Annex 1 | Annex 1 | — | New F-Block in <i>S7 F Systems Lib V1_3 SP1</i> |

A.6 Differences between the S7 F Systems Lib F-libraries

| F-Blocks | S7 F Systems Lib V1_3 SP1 | | Delta download-capable | Change from S7 F Systems Lib V1_3 to V1_3 SP1 |
|--|---------------------------|-------------------------|------------------------|---|
| | Signature | Initial value signature | | |
| F_SWC_R | Annex 1 | Annex 1 | — | New F-Block in S7 F Systems Lib V1_3 SP1 |
| SWC_MOS | Without | Without | — | New block in S7 F Systems Lib V1_3 SP1 |
| F_DEADTM | Annex 1 | Annex 1 | — | New block in S7 F Systems Lib V1_3 SP1 |
| FORCEOFF | Annex 1 | Annex 1 | — | New F-Block in S7 F Systems Lib V1_3 SP1 |
| * The change made was not relevant to the signature; therefore, the signatures have not changed. | | | | |

A.7 Run times, F-Monitoring times, and response times

Excel table S7FTIMEB.XLS contains information regarding:

- Execution times of F-Blocks in the various F-CPU's and aids for their calculation
- Maximum runtime of an F-Shutdown group
- Minimum F-Monitoring times
- Maximum response time of your F-System

This file is available for download on the Web

(<http://support.automation.siemens.com/WW/view/en/22557362>).

See also

Safety engineering in SIMATIC S7 System Manual

(<http://support.automation.siemens.com/WW/view/en/12490443>)

Checklist

Introduction

The table below contains a checklist summarizing all activities in the life cycle of a fail-safe S7 F/FH System, including requirements and rules that must be observed in the various phases.

Checklist

Key:

- Stand-alone section references refer to this documentation.
- "*SM*" refers to the "Safety Engineering in SIMATIC S7 (<http://support.automation.siemens.com/WW/view/en/12490443>)" system manual.
- "*F-SMs manual*" refers to the "Automation System S7-300 Fail-Safe Signal Modules (<http://support.automation.siemens.com/WW/view/en/19026151>)" manual.
- "*ET 200S manual*" refers to the "Distributed I/O System ET 200S, Fail-Safe Modules (<http://support.automation.siemens.com/WW/view/en/12490437>)" manual.
- "*ET 200SP manual*" refers to the "SIMATIC ET 200SP Manual Collection (<https://support.industry.siemens.com/cs/ww/en/view/84133942>)" manual collection.
- "*ET 200pro manual*" refers to the "ET 200pro Distributed I/O Device - Fail-Safe Modules (<http://support.automation.siemens.com/WW/view/en/22098524>)" manual.
- "*ET 200eco manual*" refers to the "ET 200eco Distributed I/O Station Fail-safe I/O Module (<http://support.automation.siemens.com/WW/view/en/19033850>)" manual.
- "*ET 200iSP manual*" refers to the "ET 200iSP Distributed I/O Device - Fail-safe Modules (<http://support.automation.siemens.com/WW/view/en/47357221>)" manual.

| Phase | Requirement/Rule | Reference | Check |
|--|-------------------|---|-------|
| Planning | | | |
| Requirement: A specification with the safety requirements must be available for the planned application. | Process-dependent | — | |
| Specification of the system architecture | Process-dependent | — | |
| Assignment of functions and subfunctions to the system components | Process-dependent | Section 3 <i>SM</i> , section 1.5 <i>SM</i> , section 2.4 | |

| Phase | Requirement/Rule | Reference | Check |
|--|--|--|-------|
| Selection of sensors and actuators | Requirements for actuators | <i>SM</i> , section 4.8 <i>F-SMs manual</i> , section 6.5 <i>ET 200S manual</i> , section 4.5 <i>ET 200SP manual</i> , Basic information <i>ET 200pro manual</i> , section 4.4 <i>ET 200eco manual</i> , section 5.5 <i>ET 200iSP manual</i> , section 4.5 | |
| Definition of required safety properties of the components | IEC 61508:2010 | <i>SM</i> , sections 4.7, 4.8 | |
| Configuring | | | |
| Installation of optional package | Requirements for installation | Section 4.1 | |
| Selection of S7 components | Rules for configuration | Section 3.2 <i>SM</i> , section 2.4 <i>F-SMs manual</i> , section 3 <i>ET 200S manual</i> , section 3 <i>ET 200SP manual</i> , Basic information <i>ET 200pro manual</i> , section 2 <i>ET 200eco manual</i> , section 3 <i>ET 200iSP manual</i> , sec. 2.1 | |
| Configuration of hardware | Rules for S7 F/FH Systems Verification of utilized hardware components based on Annex 1 of Certification Report | Section 5 Annex 1 of Certification Report | |
| Configuring the F-CPU | Protection level, "CPU contains safety program" Password | Sections 5.3, 6 Manual for Standard S7-400(H) | |
| Configuring the F-I/O | Settings for safety mode Configuring the monitoring times Module redundancy (optional) Defining the type of sensor interconnection/evaluation | Sections 5.2, 5.4 - 5.8 <i>SM</i> , Appendix A <i>F-SMs manual</i> , sections 3, 9, 10 <i>ET 200S manual</i> , sections 2.4, 7 <i>ET 200SP manual</i> , Device-specific information <i>ET 200pro manual</i> , sections 2.4, 8 <i>ET 200eco manual</i> , sections 3, 8 <i>ET 200iSP manual</i> , section 2.4 | |
| Programming | | | |
| Defining program design and structure | Warnings and notes on programming Verifying the utilized software components based on Annex 1 of Certification Report | Sections 7.1, 7.2, 7.6 Annex 1 of Certification Report | |
| Inserting the CFC charts | Rules for the CFC charts of the safety program | Sections 7.2.4 ff, 7.3, 7.7 | |

| Phase | Requirement/Rule | Reference | Check |
|--|--|---|-------|
| Inserting F-Runtime groups | Rules for F-Runtime groups of the safety program | Sections 7.2.7, 7.3 | |
| Defining F-Shutdown groups | Rules for F-Shutdown groups of the safety program | Section 7.2.8 | |
| Inserting and interconnecting the F blocks | Rules for F blocks | Section 7, Appendix A | |
| | Rules for F-Channel drivers and module drivers | Section 8 | |
| | Rules for interconnecting the F-block F_CYC_CO | Section 7.2.3 <i>SM</i> , Appendix A | |
| | Rules for safety-related communication between F-CPU's | Section 9 | |
| | Configuring the F-Monitoring times | Section 7.2.3, Appendix A.6 <i>SM</i> , Appendix A | |
| | Startup characteristics | Section 7.5 | |
| | Creating F block types | Section 7.7 | |
| | Passivation and reintegration | Sections 8.3, 8.4 | |
| | Data exchange between F-Shutdown groups | Section 7.8 | |
| | Data exchange with standard user program | Section 7.9 | |
| | Changing F-Parameters from one OS | Section 10 | |
| User acknowledgement | Section 7.10 | | |
| Compiling the safety program | Rules for compiling | Section 12.1 | |
| Installation | | | |
| Hardware configuration | Rules for mounting Rules for wiring | Section 14.2 <i>F-SMs manual</i> , sections 5, 6 <i>ET 200S manual</i> , sections 3, 4 <i>ET 200SP manual</i> , Basic information <i>ET 200pro manual</i> , sections 2, 3 <i>ET 200eco manual</i> , sections 3, 4 <i>ET 200iSP manual</i> , section 4 | |
| Commissioning, Testing | | | |
| Powering up | Rules for commissioning (in standard case) | Manual for Standard S7-400(H) | |
| Downloading the safety program | Rules for downloading | Sections 12.6, 12.8 | |
| Testing the safety program | Rules for deactivating safety mode Rules for testing the safety program | Sections 12.5.1, 12.7 | |
| Changing the safety program | Rules for deactivating safety mode | Section 12.5.1 | |
| | Rules for changing the safety program | Sections 12.3, 12.8 | |

| Phase | Requirement/Rule | Reference | Check |
|---|--|--|-------|
| Check of safety-related parameters | Rules for configuration | Sections 12.4, 11 <i>F-SMs manual</i> , sections 4, 9, 10 <i>ET 200S manual</i> , sections 2.4, 7 <i>ET 200SP manual</i> , Basic information <i>ET 200pro manual</i> , sections 2.4, 8 <i>ET 200eco manual</i> , sections 3, 8 <i>ET 200iSP manual</i> , section 2.4 | |
| Acceptance test | Rules and notes on the acceptance test Generating printouts | Section 13 | |
| Operation, Maintenance | | | |
| General operation | Notes on operation | Section 14 | |
| Access protection | | Section 6 | |
| Diagnostics | Responses to faults and events | Appendix A | |
| Replacement of software and hardware components | Rules for module replacement Rules for updating the operating system of the F-CPU - same as for standard system Rules for updating software components Notes on IM operating system update Notes on preventive maintenance | Section 14.2, Manual for Standard S7-400(H) | |
| Removing, disassembly | Notes for removing software components Notes for disassembling modules | Sections 4.2, 14.2 | |

Requirements for virtual environments and remote access



C.1 Summary

SIMATIC S7 F/FH Systems with S7 F Systems V6.0 and higher and Safety Matrix V6.1 SP1 and higher enable use in virtual environments for ES and OS under the following conditions.

All restrictions and notes in the corresponding releases of S7 F Systems and Safety Matrix, as well as of STEP 7 and PCS 7 continue to be valid for virtual environments and remote access.

Virtual environments

In information technology, a virtual machine refers to the emulation of a real computer system (hardware) on an abstraction layer which can execute multiple virtual machines at the same time. The abstraction layer is known as a hypervisor. Well-known manufacturers are Microsoft (Microsoft Hyper-V), VMware (VMware vSphere Hypervisor (ESXi)) and Citrix (XenServer).

A virtual environment enables, for example, very convenient test environments, simplifies the transfer of systems and saves space.

Remote Access and Control

In information technology, "remote access" designates the takeover of a graphical user interface and can be employed for different types of access. In this document, "remote access" refers to the unique access to the graphical user interface and the transfer of keyboard actions and mouse movements of an Engineering Station or Operator Station. Well-known software products include Microsoft Remote Desktop Protocol (RDP) and the RealVNC Open Source Software VNC (RFC 6143).

C.1 Summary

Recommended software requirements

SIMATIC STEP 7 and PCS 7 are released for virtual environments and remote access and can be integrated in your plant under the environment descriptions linked here.

| Products | Product news | Optional packages |
|---|--|---|
| PCS 7 V8.0 SP2: <ul style="list-style-type: none"> VMware vSphere V5.0 VMware vSphere V5.1 | https://support.industry.siemens.com/cs/ww/en/view/102378876 (https://support.industry.siemens.com/cs/ww/en/view/102378876) | <ul style="list-style-type: none"> S7 F Systems V6.1 SP2 and higher SIMATIC Safety Matrix V6.2 SP1 and higher |
| PCS 7 V8.1 and higher <ul style="list-style-type: none"> VMware vSphere V5.5 | https://support.industry.siemens.com/cs/ww/en/view/93997453 (https://support.industry.siemens.com/cs/ww/en/view/93997453) | <ul style="list-style-type: none"> S7 F Systems V6.1 SP2 and higher SIMATIC Safety Matrix V6.2 SP1 and higher |
| Service Pack 4 for STEP 7 V5.5 and higher ^{*1)} : <ul style="list-style-type: none"> VMware vSphere Hypervisor ESX(i) 5.5 VMware Workstation 10.0 VMware Player 5.02 Microsoft Windows Server 2012 Hyper-V | https://support.industry.siemens.com/cs/ww/en/view/9384200 (https://support.industry.siemens.com/cs/ww/en/view/9384200) | <ul style="list-style-type: none"> S7 F Systems V6.1 SP2 and higher SIMATIC Safety Matrix V6.2 SP1 and higher |

*1) Only configuration, programming and operation in STEP 7 Engineering.

Note

Siemens provides preconfigured virtualization solutions with its "SIMATIC Virtualization as a Service".

For more information, see the following entry: <https://support.industry.siemens.com/sc/ww/en/sc/3095>
[\(https://support.industry.siemens.com/sc/ww/en/sc/3095\)](https://support.industry.siemens.com/sc/ww/en/sc/3095)

C.2 Configuration and operation

C.2.1 Virtual environments

 WARNING**Use of virtual environments in ES/OS**

Note that a HYPERVISOR or the client software of a HYPERVISOR is not permitted to perform functions that reproduce recorded frame sequences with correct time behavior on a network with connected plants.

Ensure that this is the case when using the following functions, for example:

- Reset of captured states (snapshots) of the virtual machine (VM)
- Suspending and resuming the VM (suspend & resume)
- Replay of recorded sequences in the VMs (replay)
- Moving of VMs between hosts in productive operation (e.g. Fault Tolerance (FT))
- Digital twins of VMs in the virtual environment

If in doubt, disable these functions in the settings (HYPERVISOR administrator console).

Note

How do you use VMware vSphere Client to assign operator permissions for a virtual machine?

<https://support.industry.siemens.com/cs/ww/en/view/90142228>
(<https://support.industry.siemens.com/cs/ww/en/view/90142228>)

Note

How do you use a controller to load from a VM (VMware Player/Workstation) via a PROFIBUS/MPI CP connected via PCI or PCIe?

<https://support.industry.siemens.com/cs/ww/en/view/100450795>
(<https://support.industry.siemens.com/cs/ww/en/view/100450795>)

Note

Configure Hyper-V for Role-based Access Control

[https://technet.microsoft.com/en-us/library/dd283076\(v=ws.10\).aspx](https://technet.microsoft.com/en-us/library/dd283076(v=ws.10).aspx)
([https://technet.microsoft.com/en-us/library/dd283076\(v=ws.10\).aspx](https://technet.microsoft.com/en-us/library/dd283076(v=ws.10).aspx))

C.2.2 Remote Access and Control

 **WARNING**

Remote access from higher-level control room and Engineering Center

Make sure that the plants are clearly distinguished from other accessible plants connected on the network before you start making changes or start operation.

Examples:

- Specify optical distinguishing marks (plant designation) at your operator stations.
- The pair of numbers for SAFE_ID1 and SAFE_ID2 with SDW must be unique for all the plants accessible in the network.
- Specify unique descriptions for title and project in the properties of the Safety Matrix for all the plants connected on the network and check this before starting operation.
- Specify Active Directory access limitations in the corporate directory service and use SIMATIC Logon for accessing projects and for logging on to operator stations.

 **WARNING**

The "S7 F Systems HMI" and "Safety Matrix Viewer" functionality makes changes in the safety program during RUN mode.

As a result, the following additional safety measures are required:

- Make sure that operations that could compromise plant safety cannot be carried out. You can use the EN_SWC and EN_CHG input for this purpose, for example, by controlling it with a key-operated switch or on a process-specific basis via the safety program.
- Make sure that only authorized persons can carry out operations.

Examples:

- Control the EN_SWC or EN_CHG input with a key-operated switch.
- Control the EN_SWC or EN_CHG input with separate key-operated switches.
- Set up access protection at operator stations where process operation can be performed.

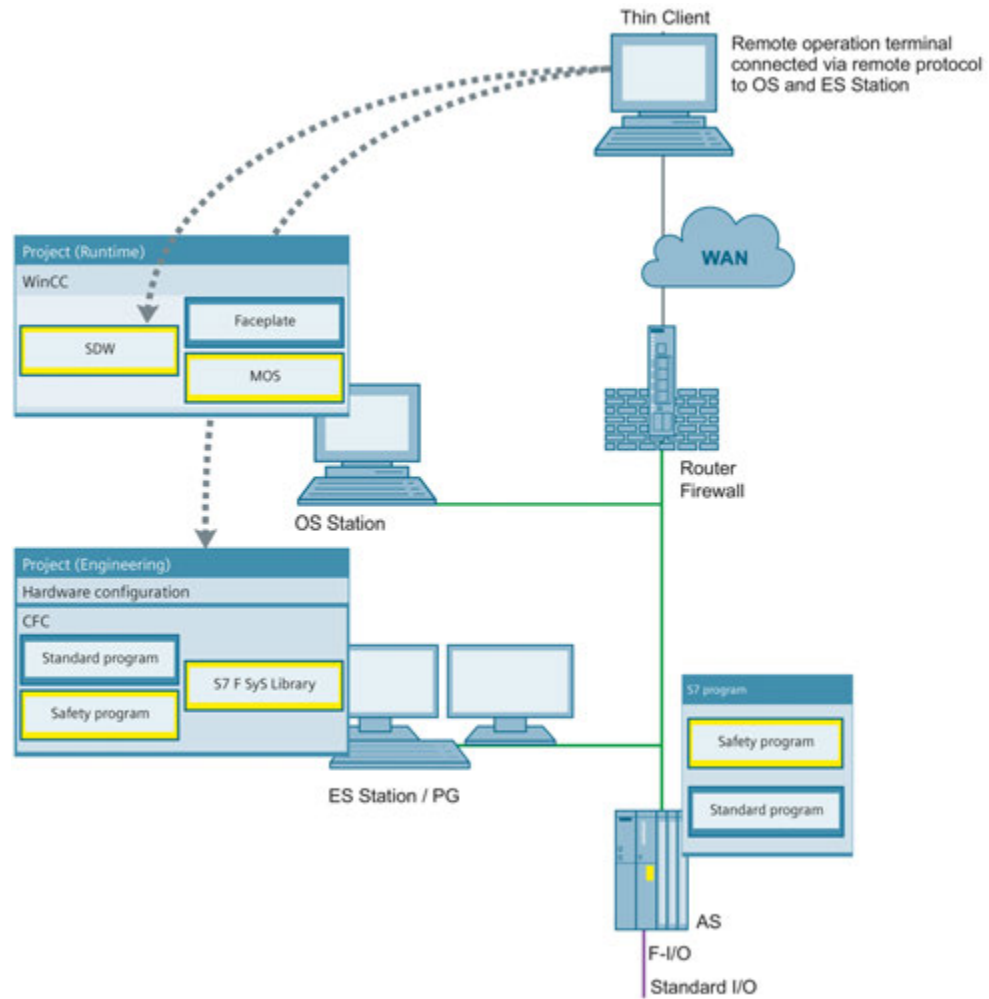
Carefully choose the persons who may have remote access to the plant and authorize them accordingly:

- Locally on the target computer "Remote Desktop User" (Workgroups)
OR
- In the Active Directory, and inherit permissions to the target computer "Remote Desktop User" (Domain).

As required, make a distinction in the WinCC authorizations between:

- Process control
- Higher process control
- Safety application control (SIF)

Figure C-1: Diagram of Engineering Station and Operator Station in projects with safety applications



ES station

Table 1: Explanation of Figure C-1

| Physical location | Installed software |
|---|--|
| At the same location as the AS station and connected to the plant/terminal bus. | SIMATIC PCS 7 (package: PCS 7 Engineering) or STEP 7 |

OS station

Table 2: Explanation of Figure C-1

| Physical location | Installed software |
|---|---|
| At the same location as the AS station and connected to the plant/terminal bus. | SIMATIC PCS 7 (package: OS Client or OS Single Station) |

Thin Client

Table 3: Explanation of Figure C-1

| Physical location | Installed software |
|--|--------------------------------|
| Not at the same location as the AS station and not connected to the plant bus. | No SIMATIC software installed. |

Note

SIMATIC Process Control System PCS 7 - PC Configuration (V8.1 SP1) - Section 5.8.2

<https://support.industry.siemens.com/cs/ww/en/view/90635791>
(<https://support.industry.siemens.com/cs/ww/en/view/90635791>)

Note

Whitepaper; Security concept PCS 7 and WinCC - Basic document

<https://support.industry.siemens.com/cs/ww/en/view/26462131>
(<https://support.industry.siemens.com/cs/ww/en/view/26462131>)

Note

How do you access WinCC and PCS 7 plants with "RealVNC"?

<https://support.industry.siemens.com/cs/ww/en/view/55422236>
(<https://support.industry.siemens.com/cs/ww/en/view/55422236>)

Note

IP-based Remote Networks

<https://support.industry.siemens.com/cs/ww/en/view/26662448>
(<https://support.industry.siemens.com/cs/ww/en/view/26662448>)

See also

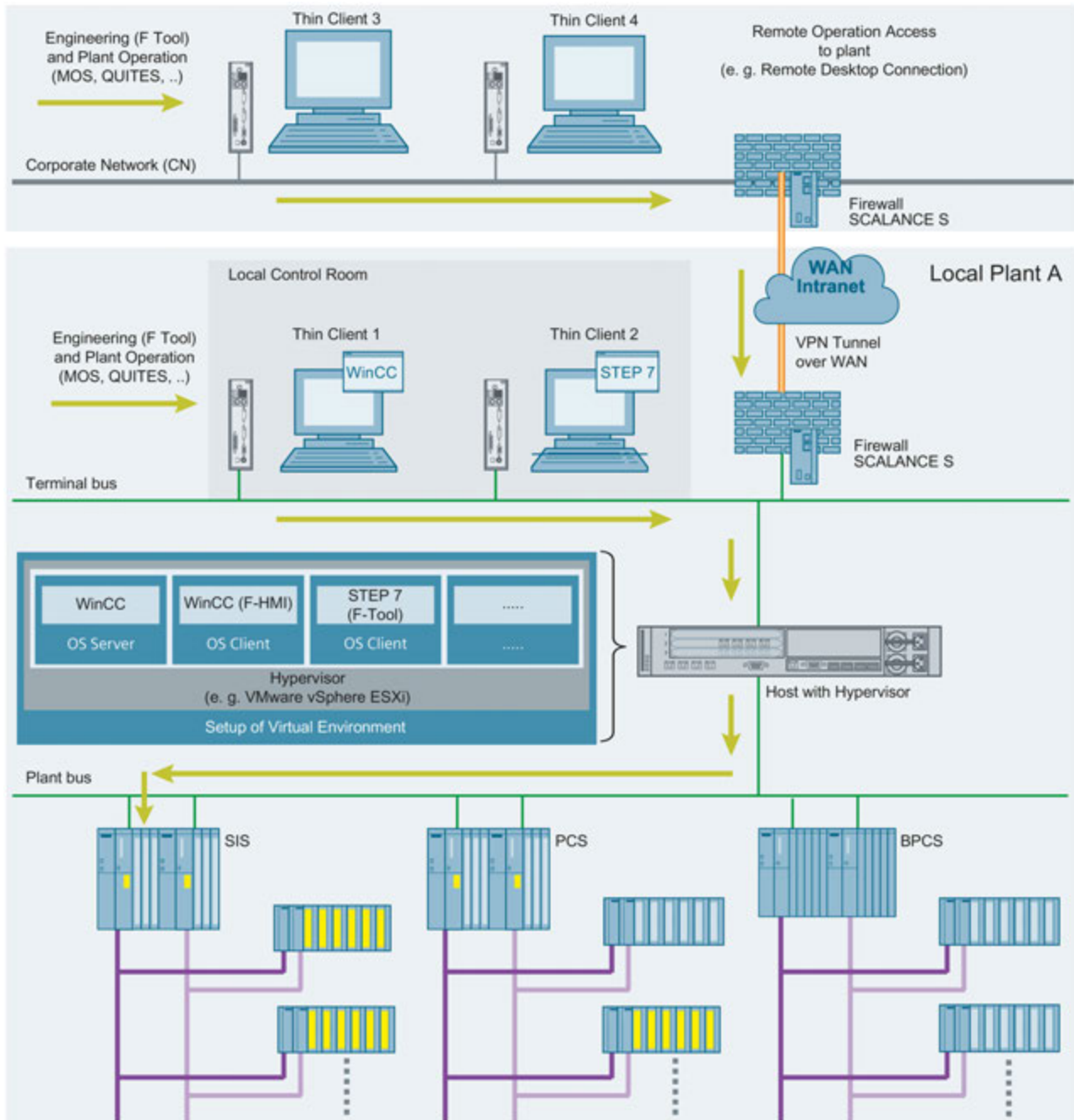
<https://support.industry.siemens.com/cs/ww/en/view/38571711>
(<https://support.industry.siemens.com/cs/ww/en/view/38571711>)

C.3 Examples of valid configurations in PCS 7

C.3.1 Example 1

The following figure shows a virtual environment for engineering and plant operation of safety applications including remote control.

Figure C-2:



C.3.2 Example 2

The following figure shows a configuration for remote access for configuration and maintenance operations as well as plant operation from higher-level control room in real and virtual environments.

Figure C-3a:

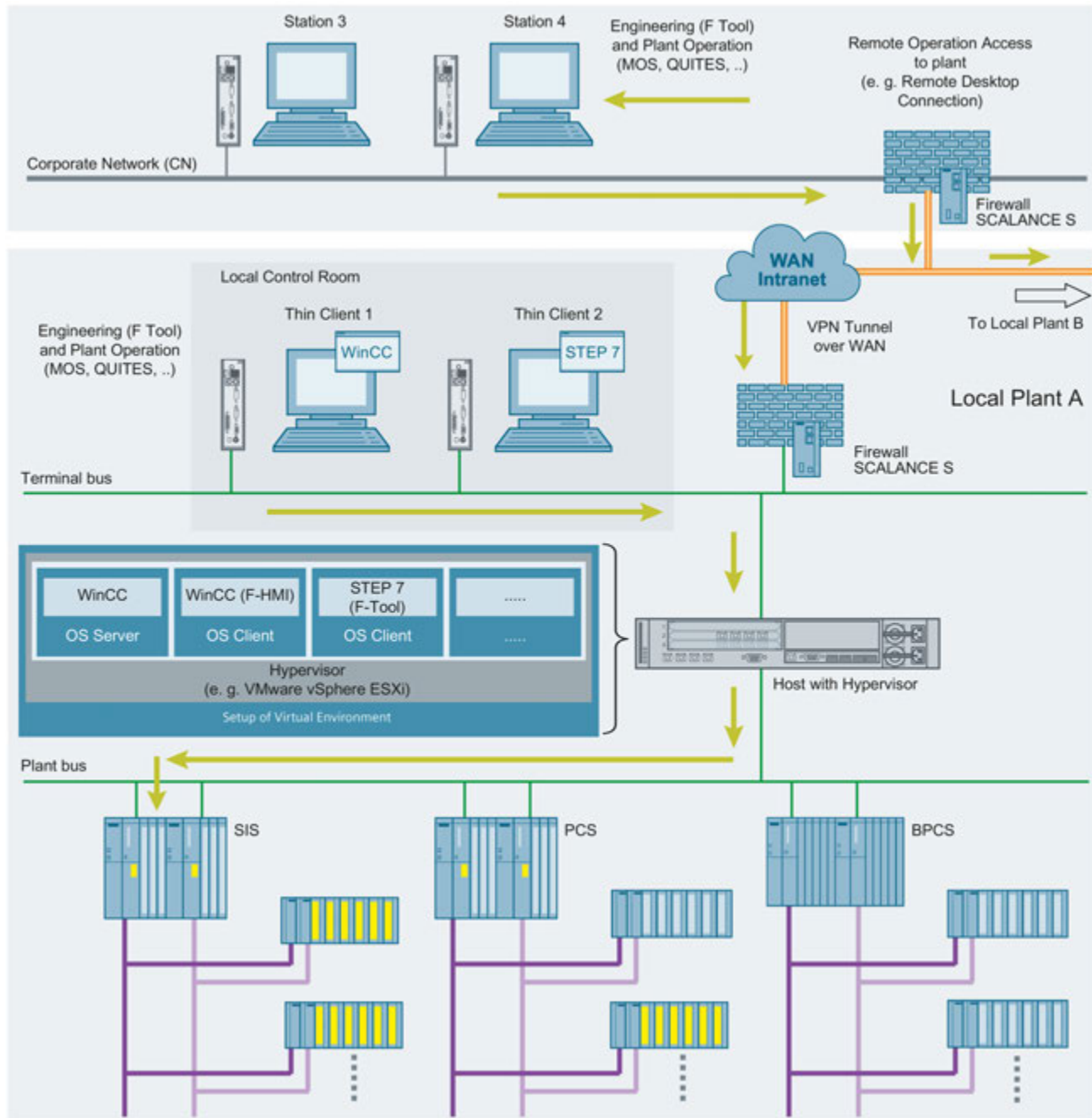
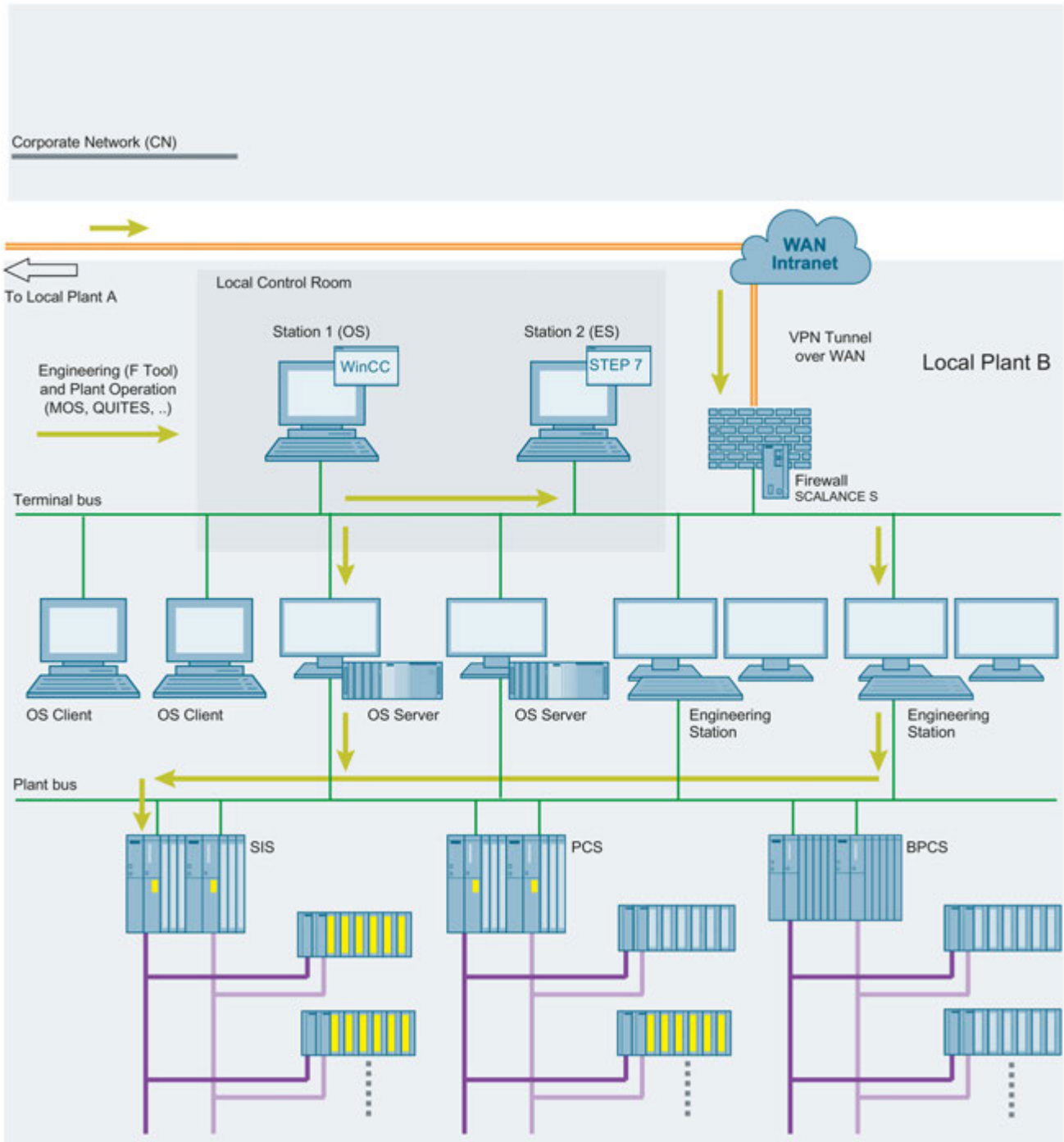


Figure C-3b:



C.4 Abbreviations and explanations of terms

| Abbreviation | Explanation of term |
|--------------|---|
| AD | Active Directory |
| BPCS | Basic Process Control System |
| CN | Corporate Network (company network/intranet) |
| ES | Engineering Station |
| LCR | Local Control Room |
| LER | Local Engineering Room |
| MOS | Maintenance Override Switch |
| OS | Operator Station |
| PCS | Process Control System |
| QUITES | Acknowledgment via ES/OS |
| ROC | Remote Operation Center (higher-level control than LCR) |
| SDW | Safety Data Write |
| SIF | Safety Instrumented Function |
| SIS | Safety Instrumented System |
| VM | Virtual Machine (guest operating system) |
| WAN | Wide Area Network |

C.5 References

| | Subject area | Link |
|------|---|--|
| \1\ | SIMATIC Industrial Software Safety Engineering in SIMATIC S7 | https://support.industry.siemens.com/cs/ww/en/view/12490443 (https://support.industry.siemens.com/cs/ww/en/view/12490443) |
| \2\ | SIMATIC Industrial Software S7 F/FH Systems - Configuring and Programming | https://support.industry.siemens.com/cs/ww/en/view/101509838 (https://support.industry.siemens.com/cs/ww/en/view/101509838) |
| \3\ | SIMATIC Industrial Software Safety Matrix | https://support.industry.siemens.com/cs/ww/en/view/100675874 (https://support.industry.siemens.com/cs/ww/en/view/100675874) |
| \4\ | SIMATIC PCS 7 technical documentation | http://w3.siemens.com/mcms/industrial-automation-systems-simatic/en/manual-overview/tech-doc-pcs7/Pages/Default.aspx (http://w3.siemens.com/mcms/industrial-automation-systems-simatic/en/manual-overview/tech-doc-pcs7/Pages/Default.aspx) |
| \5\ | SIMATIC PCS 7 OS Software Client V7.1 + SP2 and higher released for use in virtual operating environments | https://support.industry.siemens.com/cs/ww/en/view/51401737 (https://support.industry.siemens.com/cs/ww/en/view/51401737) |
| \6\ | SIMATIC Virtualization as a Service | https://support.industry.siemens.com/cs/ww/en/view/107586660 (https://support.industry.siemens.com/cs/ww/en/view/107586660) |
| \7\ | What are the options for upgrading the software of a virtualization system? | https://support.industry.siemens.com/cs/ww/en/view/103496884 (https://support.industry.siemens.com/cs/ww/en/view/103496884) |
| \8\ | VMware vSphere Documentation V5.5 | https://www.vmware.com/support/pubs/vsphere-esxi-vcenter-server-pubs.html (https://www.vmware.com/support/pubs/vsphere-esxi-vcenter-server-pubs.html) |
| \9\ | Microsoft Hyper-V | https://technet.microsoft.com/en-us/windowsserver/dd448604.aspx (https://technet.microsoft.com/en-us/windowsserver/dd448604.aspx) |
| \10\ | XenServer Documentation Index | http://docs.vmd.citrix.com/XenServer/6.5.0/1.0/en_gb/ (http://docs.vmd.citrix.com/XenServer/6.5.0/1.0/en_gb/) |

C.5 References

Glossary

1oo1 evaluation

Type of -> sensor evaluation: In the case of 1oo1 evaluation, a non-redundant sensor is connected via one channel to the -> F-I/O.

1oo2 evaluation

Type of -> sensor evaluation: In the case of 1oo2 evaluation, two input channels are occupied either by one two-channel sensor or by two single-channel sensors. The input signals are compared internally for equality (equivalence) or inequality (nonequivalence).

Access protection

-> Fail-safe systems must be protected against dangerous, unauthorized access. Access protection for F-Systems is implemented by assigning two passwords (for the -> F-CPU and for the -> safety program).

Bypass

Bypass function that is normally used for maintenance purposes (e.g., for checking effect logic, replacing a sensor).

Category

Category according to ISO 13849-1:2015 or EN ISO 13849-1:2015
S7 F Systems can be used in -> safety mode up to Category 4.

Channel fault

Channel-specific fault, such as a wire break or a short-circuit

Collective signatures

Collective signatures uniquely identify a particular state of the -> safety program. They are important for the preliminary acceptance test of the safety program, e.g., by experts.

CRC

Cyclic Redundancy Check -> CRC signature

CRC signature

The validity of the process data in the -> safety message frame, the accuracy of the assigned address references, and the safety-related parameters are ensured by means of a CRC signature contained in the -> safety message frame.

Deactivated safety mode

Deactivated safety mode is the temporary deactivation of -> safety mode for test purposes, commissioning, etc.

Whenever safety mode is deactivated, the safety of the system must be ensured by other organizational measures, such as operation monitoring and manual safety shutdown.

Depassivation

-> Reintegration

Discrepancy time

Assignable time for the discrepancy analysis. If the discrepancy time is set too high, the fault detection time and fault reaction time are extended unnecessarily. If the discrepancy time is set too low, availability is decreased unnecessarily because a discrepancy error is detected when, in reality, no error exists.

ES

Engineering System (ES): Configuration system that enables convenient, visual adaptation of the process control system to the task at hand.

Fail-safe DP standard slaves

Fail-safe DP standard slaves are standard slaves that are operated on PROFIBUS with the DP protocol. They must behave in accordance with IEC 61784-1:2002 Ed1 CP 3/1 and the PROFIsafe bus profile. A GSD file is used for your configuration.

Fail-safe I/O modules

ET 200eco modules that can be used for safety-related operation (in -> safety mode). These modules are equipped with integrated -> safety functions. They behave according to IEC 61784-1:2002 Ed1 CP 3/1 and the PROFIsafe bus profile.

Fail-safe IO standard devices

Fail-safe IO standard devices are standard devices that are operated on PROFINET IO.

A GSDML file is used to configure them.

Fail-safe modules

ET 200S modules that can be used for safety-related operation (-> safety mode) in the ET 200S or ET 200pro distributed I/O system. These modules are equipped with integrated -> safety functions. They behave according to IEC 61784-1:2002 Ed1 CP 3/1 and CP 3/3 and the PROFIsafe bus profile.

Fail-safe PA field devices

Fail-safe PA field devices are field devices that are operated on PROFIBUS with the PA protocol. They must behave in accordance with IEC 61784-1:2002 Ed1 CP 3/2 and the PROFIsafe bus profile. A GSD file is used for your configuration.

Fail-safe systems

Fail-safe systems (F-Systems) are systems that remain in a -> safe state or immediately switch to another safe state when particular failures occur.

Fault reaction function

-> User safety function

F-block type

F-block types are ready-made program sections that can be used in a CFC chart (e.g., fail-safe multiplexer F_MUX2_R, etc.). Block instances are generated on insertion. Any number of block instances can be created by one F-block type.

The F-block type specifies the characteristics (algorithm) for all applications of this type. The name of the F-block type is specified in the symbol table.

F-blocks

The following fail-safe blocks are designated as F-Blocks:

- Blocks selected by the user from an F-Library
- Blocks that are automatically added in the -> safety program

F-CPU

An F-CPU is a central processing unit with fail-safe capability that is permitted for use in *S7 F Systems*. For *S7 F Systems*, the F-Runtime license allows the user to operate the central processing unit as an F-CPU. That is, a -> safety program can be run on it. A -> standard user program can also be run in the F-CPU.

F-Cycle time

Cyclic interrupt time for OBs with -> F-Runtime groups

F-Data type

The standard user program and -> safety program use different data formats. Safety-related F-Data types are used in the safety program.

F-I/O

Group designation for fail-safe inputs and outputs available in *SIMATIC S7* for integration in *S7 F Systems*, among others.

You can find additional information on available F-I/O for *S7 F Systems* in section "Overview on configuring".

F-Runtime group

When the -> safety program is created, the -> F blocks cannot be inserted directly into tasks/OBs; rather, they must be inserted into F-Runtime groups. The -> safety program consists of multiple F-Runtime groups.

F-Shutdown groups

F-Shutdown groups contain one or more -> F-Runtime groups. F-Runtime group communication blocks between the -> F blocks in various F-Runtime groups, all of which are assigned to one F-Shutdown group, are not required. If an error is detected in an F-Shutdown group, this F-Shutdown group is shut down. Additional F-Shutdown groups are shut down according to the configuration of F_SHUTDN.

F-SMs

S7-300 fail-safe signal modules that can be used for safety-related operation (in -> safety mode) as centralized modules in an S7-300 or as distributed modules in the ET 200M distributed I/O system. F-SMs are equipped with integrated -> safety functions.

F-Startup

An F-Startup is a restart following an F-STOP or an F-CPU STOP. *S7 F Systems* do not distinguish between a cold restart and warm restart of the F-CPU.

F-Systems

Fail-safe systems

Full shutdown

All F-blocks of the entire F-CPU are shut down. Initially, the F-Shutdown group in which the error was detected is shut down. All other F-Shutdown groups are then shut down within a period of time equal to twice the F-Monitoring time you assigned for the slowest OB.

Master-reserve switchover

In S7 FH Systems, a master/reserve switchover is triggered when the master goes to F-STOP mode. That is, the system switches from the master CPU to the reserve CPU.

Module redundancy

The module and a second identical module are operated in redundant mode in order to enhance availability.

OS

Operator Station (OS): A configurable operator station used to operate and monitor machines and systems.

Partial shutdown

Only the F-shutdown group in which the error was detected is shut down.

Passivation

Passivation of digital output channels means that the outputs are de-energized.

Digital input channels are passivated when the inputs transmit a value of "0" to the F-CPU (by means of the fail-safe drivers), irrespective of the current process signal.

Analog input channels are passivated when the inputs transmit a fail-safe value or the last valid value to the F-CPU (by means of the fail-safe drivers), irrespective of the current process signal.

Process safety time

The process safety time of a process is the time interval during which the process can be left on its own without risk to life and limb of the operating personnel or damage to the environment.

Within the process safety time, any type of F-System process control is tolerated. That is, during this time, the -> F-System can control its process incorrectly or it can even exercise no control at all. The process safety time depends on the process type and must be determined on a case-by-case basis.

PROFIsafe

Safety-related bus profile of PROFIBUS DP/PA and PROFINET IO for communication between the -> Safety program and the -> F-I/O in an > F-System.

Proof-test interval

Period after which a component must be forced to fail-safe state, that is, it is either replaced with an unused component, or is proven faultless.

Redundancy, availability-enhancing

Multiple instances of components with the goal of maintaining component function even in the event of hardware faults.

Redundancy, safety-enhancing

Multiple availability of components with the focus set on exposing hardware faults based on comparison; for example, → 1oo2 evaluation in fail-safe signal modules.

Reintegration

Switchover from fail-safe values (0) to process data (reintegration of an F-I/O module) occurs automatically or, alternatively, only after user acknowledgment at the F-Channel driver.

The reintegration method depends on the following:

- Cause of passivation of the F-I/O or channels of the F-I/O
- Parameter assignment for the F-Channel driver

For an F-I/O with inputs, the process values pending at the fail-safe inputs are provided again at the output of the F-Channel driver after reintegration. For an F-I/O with outputs, the F-System again transfers the output values pending at the input of the F-Channel driver to the fail-safe outputs.

S7 F Systems RT License (Copy License)

Formal authorization for use of the CPU as an F-CPU for S7 F/FH Systems.

S7-PLCSIM

The *S7-PLCSIM* application enables you to execute and test your S7 program on a simulated automation system on your ES/OS. Because the simulation takes place entirely in STEP 7, you do not require any hardware (CPU, F-CPU, I/O).

Safe state

The basic principle of the safety concept in -> fail-safe systems is the existence of a safe state for all process variables. For digital -> F-I/O, the safe state is always the value "0".

Safety class

Safety level (Safety Integrity Level) SIL according to IEC 61508. The higher the Safety Integrity Level, the more rigid the measures for prevention of systematic faults and for management of systematic faults and random hardware failures.

S7 F Systems can be used in safety mode up to safety class SIL3.

Safety function

Mechanism built into the -> F-CPU and -> F-I/O that allows them to be used in -> fail-safe systems.

According to IEC 61508, function implemented by a safety device in order to maintain the system in a -> safe state or to place it into a safe state in the event of a particular fault (-> user safety function).

Safety message frame

In -> safety mode, data are transferred between the -> F-CPU and -> F-I/O or between the F-CPU's in safety-related CPU-CPU communication in a safety message frame.

Safety mode

1. Safety mode is the operating mode of the -> F-I/O that allows -> safety-related communication by means of -> safety message frames.
2. Operating mode of the safety program. In safety mode of the safety program, all safety mechanisms for fault detection and fault reaction are activated. In safety mode, the safety program cannot be modified during operation. Safety mode can be deactivated by the user (-> deactivated safety mode).

Safety program

Safety-related user program

Safety protocol

-> Safety message frame

Safety-related communication

Communication used to exchange fail-safe data.

Sensor evaluation

There are two types of sensor evaluation:

- 1oo1 evaluation – sensor signal is read in once
- 1oo2 evaluation - sensor signal is read in twice by the same ->F-I/O and compared internally

Signature

-> Collective signatures

Standard communication

Communication used to exchange non-safety-related data.

Standard mode

Operating mode of -> F-I/O in which -> safety-related communication by means of -> safety message frames is not possible, but rather only -> standard communication.

Standard user program

Non-safety-related user program

User safety function

The -> safety function for the process can be provided through a user safety function or a -> fault reaction function. The user only has to program the user safety function. In the event of a fault in which the -> F-System can no longer execute its actual user safety function, it will execute the fault reaction function: For example, the associated outputs are deactivated and the -> F-CPU switches to STOP mode if necessary.

Index

A

- Acceptance
 - F-block types, 222
- Acceptance test
 - Overview, 213
- Access protection, 69
- Access to F-I/O, 109
- Activating safety mode, 195
- Addressing
 - PROFIsafe, 47
- AND logic operation, 232
- Application case, 128, 130, 132, 134, 136, 138
- Assigning parameters
 - of F-CPU, 47
- Authorizations
 - Default, 141, 167
- Automatically inserted F-Blocks, 89

B

- Backup of the safety program, 217
- Behavior of F-Cycle time monitoring, 81
- Binary selection, 235, 236
- Block I/O
 - Fail-safe, 203
- Button
 - Library version, 182
 - Refresh, 184
 - Safety mode, 195

C

- Changing non-interconnected inputs in CFC test mode, 202
- Checklist, 461
- CiR
 - Adding fail-safe I/O, 65
 - Configuring, 65
 - Deleting fail-safe I/O, 66
 - Synchronization time, 65
- Collective signature, 81
- Communication
 - Configuring via S7 connections, 115
 - Programming from safety program to standard user program, 103

- Programming from standard user program to safety program, 104
 - Via S7 connections, 115
- COMPLEM component, 230
- Components of S7 F/FH systems, 26
- Compression, 83
- Configuration
 - CiR, 64
- Configuring
 - Overview, 45
 - Redundant fail-safe signal modules, 62
 - Safety-related communication via S7 connections, 115
 - with GSD/GSDML file, 56
- Confirmer, 162, 168, 172
 - Confirming the change, 175
- ConfirmerAuthorization, 168, 172
- Connection table, 115
- Continuous Function Chart (CFC)
 - Notes, 83
- Conversion
 - BOOL to F_BOOL, 288
 - F_BOOL to BOOL, 304
 - F_REAL to REAL, 304
 - REAL to F_REAL, 289
- Conversion blocks, 103
- Creating F-Block types, 95
- Creating the safety program, 80
- Cyclic interrupt, 78, 81

D

- DATA component, 230
- Data exchange
 - between standard user program and safety program, 103
 - programming between F-shutdown groups, 101
- Deactivating safety mode, 195
- Determining the program structure, 81
- Dialog
 - Safety program, 180
- Displaying Help, 30
- Downloading
 - Entire safety program, 199
 - in RUN mode, 198
 - S7 program, 198
- Downloading changes, 198

E

ET 200SP, 53
 Exclusive OR logic operation, 234

F

F block types

- fail-safe, 95
- F_1oo2_R, 396, 396
- F_1oo2AI, 272
- F_2oo3_R, 395
- F_2oo3AI, 267
- F_2oo3DI, 265
- F_2OUT3, 235
- F_ABS_R, 389
- F_ADD_R, 387
- F_AND4, 232
- F_AVEX_R, 393
- F_BO_FBO, 103, 104, 288
- F_CH_AI, 336
- F_CH_BI, 310
- F_CH_BO, 314
- F_CH_DI, 328
- F_CH_DII, 353
- F_CH_DIO, 358
- F_CH_DO, 332
- F_CH_RI, 362
- F_CHG_BO, 161, 283, 299
- F_CHG_R, 161, 285, 293
- F_CMP_R, 262
- F_CTUD, 374
- F_CYC_CO, 81
- F_destination_address, 49, 51, 53
 - Assigning, 55
 - Changing, 55
- F_DIV_R, 388
- F_F_TRIG, 385
- F_FBO_BO, 103, 104, 304
- F_FL_FR, 292, 402, 404
- F_FL_I, 103, 104, 305
- F_FR_R, 103, 104, 304
- F_FTI_TI, 103, 104, 305
- F_I_FI, 292
- F_LIM_HL, 263
- F_LIM_I, 398
- F_LIM_LL, 264
- F_LIM_R, 391
- F_LIM_TI, 384
- F_MAX3_R, 389
- F_MID3_R, 390
- F_MIN3_R, 391

- F_MOV_R, 399
- F_MUL_R, 388
- F_MUX16R, 401
- F_MUX2_R, 401
- F_NOT, 235
- F_OR4, 233
- F_PA_AI, 319
- F_PA_DI, 323
- F_PS_12, 426
- F_PSG_M, 85, 371
- F_QUITES, 289
- F_R_BO, 101
- F_R_FR, 103, 104, 289
- F_R_R, 101, 370
- F_R_TRIG, 385
- F_RCVBO, 117, 117, 242
- F_RCVR, 117, 117, 249
- F_RDS_BO, 117
- F_REPCYC, 379
- F_ROT, 382
- F_RS_FF, 372
- F_S_BO, 101, 367
- F_S_R, 101, 369
- F_SDS_BO, 117
- F_SENDBO, 117, 117, 238, 254, 258
- F_SENDR, 117, 117, 246
- F_SHUTDOWN, 422
- F_SMP_AV, 394
- F_source_address, 51, 53
 - Assigning, 55
 - Changing, 55
- F_SQRT, 392
- F_SR_FF, 373
- F_START, 371
- F_SUB_R, 387
- F_SWC_CB, 276
- F_SWC_CR, 278
- F_SWC_P, 282
- F_TI_FTI, 291
- F_TOF, 378
- F_TON, 376
- F_TP, 375
- F_XOR2, 234
- F_XOUTY, 236
- Fail-safe PA field device
 - F-channel drivers, 319, 323
- Fail-safe systems, 23, 69
 - Access protection, 69
- Fail-safe user times, 376, 377, 379
- F-block types
 - Acceptance, 222
 - Creating, 97

- F-block types
 - Integrate F-parameters in printout, 99
 - F-Block types
 - Modify, 99
 - F-blocks
 - Data conversion, 274
 - F-channel drivers, 310
 - Voter blocks for inputs of REAL and BOOL data type, 265
 - F-Blocks, 81
 - Arithmetic Blocks of the INT Data Type, 398
 - Arithmetic Blocks of the REAL Data Type, 386
 - Assigning parameters, 86
 - Automatically inserted, 89
 - F-Control blocks, 414
 - Flip-Flops, 372
 - F-System blocks, 367
 - IEC pulse and counter blocks, 374
 - Inserting, 86
 - Interconnecting, 86
 - Logic blocks of data type BOOL, 232
 - Multiplex Blocks, 399, 402, 410
 - Names, 86
 - Pulse Blocks, 379
 - Rules, 86
 - Rules for interconnecting, 86
 - Runtime sequence, 87
 - F-channel drivers, 109, 310
 - for fail-safe DP standard slaves/IO standard devices, 310, 314, 353, 358, 362
 - for fail-safe PA field device, 319, 323
 - F-Control blocks, 414
 - F-conversion blocks, 103, 104
 - F-CPU password, 72
 - F-Cycle time:Changing, 81
 - F-Data types, 86, 230
 - F-destination_address
 - 'Naming' function, 53
 - F-driver blocks
 - F-channel drivers, 109
 - F-module drivers, 109
 - F-I/O
 - Access, 109
 - Use of a process image partition, 109
 - Fiber-optic cables, 226
 - F-module drivers, 109
 - F-Monitoring times
 - Calculating, 64
 - Reducing, 64
 - F-Runtime groups, 78
 - Sampling rate, 86
 - F-shutdown groups
 - Combining, 85
 - Maximum number, 81, 81
 - F-source_address, 53
 - F-startup, 91
 - Restart/startup protection, 92
 - F-STOP
 - Ending, 94
 - Full shutdown, 93
 - Partial shutdown, 93
 - Types, 93
 - F-System blocks, 367
 - Full shutdown, 181
 - Function test of the safety program, 219
- ## G
- Group diagnostics, 50
- ## H
- Hardware components, 26
 - Hardware configuration data
 - Checking, 214
 - H-CiR
 - Adding or removing F-I/O, 67
 - H-systems, 81
 - HW configuration data
 - Printing, 214
- ## I
- Initial acceptance
 - of a safety program, 219
 - Initial acceptance test, 214
 - of a safety program, 214
 - Initiator, 162, 168, 172
 - Initiating a change, 173
 - InitiatorAuthorization, 168, 172
 - Inputs
 - Non-interconnected, 202
 - Installation, 30
 - Optional package, 32
- ## L
- Library version, 182
 - License key, 30
 - Life cycle of fail-safe automation systems, 461

Limit

- Lower limit violation, 264
- Upper limit violation, 263

Local ID of the S7 connection, 115

M

Memory card, 198

N

Non-interconnected inputs, 202

O

OB 100, 90

OB 3x, 78, 81

- Cycle time, 48

Operator

- Confirmer, 162
- Initiator, 162

Operator function, 128, 130, 132, 134, 136, 138

- Based on 'Secure Write Command++', 125
- Blocks and components, 123
- Change process values, 125
- Configuring, 126
- Configuring faceplates, 140
- Fail-safe acknowledgment, 125
- Maintenance Override, 125
- Multiple operator inputs per shutdown group, 128
- Principle of configuration and operation, 126
- Secure Write Command, 123
- Use of a keyswitch, 127
- User authorizations, 141

Operator input

- Change process value with two operators, 146
- Changing a process value with only one operator, 149
- Fail-safe acknowledgment with only one operator, 159
- Fail-safe acknowledgment with two operators, 156
- Maintenance Override with only one operator, 156

Operator input

- Maintenance Override with two operators, 150

Operator station (OS), 161

OR logic operation, 233

OS

- Client, 145, 170
- Operator station, 161

P

Partial shutdown, 93, 181

Partner ID of the S7 connection, 115

Passivation

- F-I/O with outputs, 225

Password, 69, 198

- F-CPU, 47

Password for safety program

- Changing, 73
- Dialog, 182
- Revoking access permission, 75
- Setting up, 73

Performance improvement, 81

Placing and interconnecting F blocks, 80

PLCSIM, 200

Preliminary acceptance test of configuration of the F-I/O, 214

Preventive maintenance (Proof Test), 225

Printing

- Hardware configuration data, 214
- of a safety program, 192

Priority class, 81

Process image partition, 109

PROFIsafe

- Addressing, 47
- PROFIsafe address, 51
- Assignment rules, 52
- F_destination_address, 51
- F_source_address, 51

PROFIsafe stations, 223

Project structure, 80

Proof Test, 225

R

Receiving

- F_BOOL data, 242
- F_REAL data, 249

Redundant fail-safe signal modules

- Configuring, 62

Refresh, 184

Remote access, 465

Removing, 31

Repair, 225

- Duration, 225

Replacement

- Hardware components, 225
- Software components, 225

Requirements

- Software, 29

Requirements, installation, 32

- Response time
 - Change, 34
- Restart/startup protection, 92
- Rules
 - for changing non-interconnected inputs, 202
 - for downloading, 198
 - For F-systems, 46
 - For interconnecting F-Blocks, 86
 - For operation, 223
 - for testing, 200
 - for the data exchange between F-shutdown groups, 101
 - For the program structure, 81
- Runtime sequence
 - Defining, 88
 - F-Blocks, 87
- S**
- S7 F Systems
 - Program structure, 78
 - Removing, 226
- S7 F Systems optional package, 27
 - Components, 26
 - Installation, 32
 - Removing, 31
 - Version, 218
- S7 F Systems RT License (Copy License), 30
- S7 FH
 - Both F-CPU's simultaneously as master, 223
 - Fiber-optic cables between synchronization modules, 223
- S7 program
 - Compiling, 179
- SAFE_ID1 and SAFE_ID2
 - Safety Data Write, 297, 302
- Safety data format, 230
- Safety Data Write, 161, 293, 299
 - Basic procedure, 163
 - Configuring faceplates, 166
 - F-Parameters, 170
 - Inserting F-Blocks, 163
 - MAXDELTA, 293
 - Operator types, 162
 - Safety Data Write transaction, 161
 - TIMEOUT, 293
 - User authorizations, 168
- Safety information for programming, 82
- Safety Integrity Level (SIL), 23
- Safety level, 23
- Safety mode
 - Activating, 196
 - Deactivating, 195
- Safety program, 28
 - Backup, 217
 - Comparing, 185
 - Downloading, 198
 - Function test for initial acceptance, 219
 - Initial acceptance test, 214
 - on the memory card, 198
 - Printing, 192
 - Program structure (S7 F Systems), 78
 - Testing, 200
- Safety program dialog, 180
- Safety-related communication via S7 connections, 115
 - Configuring, 115
- Safety-related parameters, 214
- Secure Write Command, (See operator function), 125
- Send
 - F_BOOL data, 258
- Sending
 - F_BOOL data, 238, 254
 - F_REAL data, 246
- Setting up an access permission for the F-CPU, 71
- Shutdown behavior, 181
- Signature, 81, 83
- Simulation
 - of a safety program, 200
 - of PROFIsafe stations, 223
 - with S7-PLCSIM, 200
- Software
 - Components, 27
 - Requirements, 29
- Structure element
 - Selection, 86
- SWC_CHG, 306
- SWC_QOS, 308
- Symbolic names, 50
- T**
- Task, 81
- Testing
 - Offline, 200
 - Rules, 200
- Transaction
 - with only one operator, 177
 - with two operators, 172

U

Usage authorization, 30

User authorizations for operators, 141, 168

User times

 Inaccuracy, 376, 377, 379

V

Version

 S7 F Systems optional package, 218

Virtual environment, 465