

SIEMENS

SIMATIC

Process Control System PCS 7 Compendium Part B - Process Safety

Operating Manual

Preface

Installing the fail-safe system

1

Advanced PCS 7 ES settings

2

Configuring S7 F/FH
hardware

3

Configuring the safety
program

4

Configuration with Safety
Matrix

5

Changes, tracking changes,
and acceptance

6

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
with a safety alert symbol, indicates that minor personal injury can result if proper precautions are not taken.
CAUTION
without a safety alert symbol, indicates that property damage can result if proper precautions are not taken.
NOTICE
indicates that an unintended result or situation can occur if the corresponding information is not taken into account.

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 device/system may only be set up and used in conjunction with this documentation. Commissioning and operation of a device/system may only be performed by **qualified personnel**. Within the context of the safety notes in this documentation qualified persons are defined as persons who are authorized to commission, ground and label devices, systems and circuits in accordance with established safety practices and standards.

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 adhered to. The information in the relevant documentation must be observed.

Trademarks

All names identified by ® are registered trademarks of the 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.

Preface

Subject of the manual

SIMATIC PCS 7, as a distinctly open system, can be flexibly adapted to a wide range of customer needs. The system software provides the project engineer with a great deal of freedom in terms of the project configuration, as well as in the design of the program and the visualization.

Experience has shown that subsequent modernization or plant expansion work is made much easier if the project is configured "in conformance with PCS 7" as far as possible right from the start. This means that certain basic rules absolutely must be adhered to in order to ensure that the system functions provided will offer optimum usability in the future.

This manual serves as a compendium in addition to the product documentation covering SIMATIC PCS 7. The basic sequences involved in creating and assigning parameters to the project are described in the form of operating instructions with numerous illustrations.

The compendium directly reflects the recommended method for configuration, which is based on the results of a great deal of practical experience. The description does not cover the application itself; rather, it relates to handling the project and the parameter settings of the components it contains.

The compendium is divided into three parts:

- Part A: Configuration guidelines
- Part B: Process safety
- Part C: Technical functions with SFC types

Validity

The configuration guidelines are valid for PCS 7 V7.0, SP1 and higher, taking the differences between that and PCS 7 V6.1, SP1 and higher, into account.

Subject of Part B - Process safety

Part B is dedicated to the implementation of the fail-safe part of an S7 program.

It examines the following F software components:

- S7 F Systems
- Safety Matrix Engineering Tool

Additional support

If this manual does not contain the answers to any questions you may have about how to use the products described, please contact your local Siemens representative.

You can locate your contact at: <http://www.siemens.com/automation/partner>

You can find a guide to the technical documentation available for individual SIMATIC products and systems at: <http://www.siemens.de/simatic-tech-doku-portal>

The online catalog and online ordering system are available at:
<http://mall.automation.siemens.com/>

Training center

Siemens offers a number of training courses to familiarize you with the SIMATIC PCS 7 process control system. Contact your regional training center or the main training center in Nuremberg, Germany: <http://www.sitrain.com>

Technical support

You can contact technical support for all Industry Automation and Drive Technology products using the Support Request web form: <http://www.siemens.de/automation/support-request>

More information about our technical support services is available on the Internet at:
<http://www.siemens.de/automation/service>

Service & Support on the Internet

In addition to our documentation options, our expertise is also available to you online.
<http://www.siemens.com/automation/service&support>

Here you will be able to access:

- The newsletter, which will keep you constantly up-to-date with the latest information about our products
- The right documents for you via the search facility in our Service & Support portal
- A forum that provides users and specialists with an international platform for exchanging experiences
- Your local contact partner for Industry Automation and Drive Technology
- Information about local service, repairs, spare parts The "Services" section offers even more options.

Table of contents

	Preface	3
1	Installing the fail-safe system	9
1.1	Software components	9
1.2	Installation on the PCS 7 engineering station (ES).....	10
1.2.1	Installing S7 F Systems	10
1.2.2	Installing Safety Matrix.....	10
1.3	Installation on the PCS 7 operator station (OS).....	11
1.4	OS client installation	11
1.4.1	Installing S7 F Systems	11
1.4.2	Installing Safety Matrix.....	11
2	Advanced PCS 7 ES settings	13
2.1	Access protection.....	13
2.1.1	General measures	13
2.1.2	Access protection with SIMATIC Logon	14
2.2	CFC settings for compiling and downloading	15
3	Configuring S7 F/FH hardware	17
3.1	Adapting CPU parameters (single F-system)	17
3.1.1	Password and access protection	17
3.1.2	Cyclic interrupts.....	18
3.1.3	Diagnostics/Clock	24
3.1.4	H parameters	25
3.2	Adapting CPU parameters (fault-tolerant F system).....	26
3.2.1	Reaction to RAM/PIQ comparison error	26
3.2.2	Cyclic interrupt OB with special handling.....	28
3.2.3	Monitoring times.....	29
3.2.4	Calculating monitoring times.....	30
3.3	Communications module parameters/Networks	34
3.4	Setting system parameters for F-signal modules	34
3.4.1	Operating mode	34
3.4.2	PROFsafe addresses.....	35
3.4.3	Module parameters - general.....	36
3.4.4	Activating channels	37
3.4.5	Assigning parameters for the SM326; DI 8 x NAMUR/SM326; DI 24 x DC 24V	39
3.4.6	Assigning parameters for the SM326; DO 10 x DC 24V/2A	41
3.4.7	Assigning parameters for the SM326; DO 8 x DC24 V/2A PM.....	42
3.4.8	Assigning parameters for the SM336; AI 6 x 13Bit.....	43
3.4.9	Assigning parameters for the SM336; F-AI 6 x 0/4...20mA HART	46
3.5	Configuring redundant F-signal modules	49
3.6	Marshaled Termination Assemblies (MTAs).....	51
3.7	"Wiring and Voting" architectures for ET 200M	52
3.7.1	Voting with F-DI	52
3.7.2	Voting with F-AI.....	54

4	Configuring the safety program.....	55
4.1	Introduction	55
4.2	Creating the safety program.....	57
4.2.1	Defining the program structure	58
4.2.2	Creating CFC charts	58
4.2.3	Assigning parameters to and interconnecting F-blocks	59
4.2.4	Run sequence of F-blocks	60
4.2.5	F-runtime groups.....	62
4.2.6	F-shutdown groups	65
4.2.7	Data exchange between the F user program and standard user program.....	68
4.2.8	How F-blocks with floating-point operations respond to number range overflows	69
4.3	Configuring fail-safe AS-AS communication	72
4.3.1	Configuring S7 connections	73
4.3.2	Configuring F-communications blocks	74
4.4	F-STOP	77
4.4.1	Complete shutdown	77
4.4.2	Partial shutdown.....	77
4.4.3	Parameter assignment for shutdown behavior	78
4.4.4	Causes of errors.....	79
4.4.5	Execution of an F-STOP in S7 F/FH systems.....	80
4.4.6	Exiting an F-STOP	81
4.5	F startup and (re)start protection	81
4.5.1	F-startup	81
4.5.2	(Re)start protection	82
4.6	I/O access via F driver blocks	83
4.7	Passivation and reintegration of input/output modules	84
4.7.1	Passivation - general.....	84
4.7.2	Group passivation	85
4.7.3	Reintegration following elimination of errors.....	85
4.7.4	Automatic reintegration on channel error.....	86
4.7.5	Programming reintegration following user acknowledgment	87
4.8	Compiling the F-program	95
4.8.1	Password protection for the safety program	95
4.8.2	Parameterizing the maximum F cycle monitoring time	96
4.8.3	Compiling the S7 program	97
4.9	Safety mode and downloading the safety program	99
4.9.1	Information on safety mode.....	99
4.9.2	Deactivating safety mode.....	99
4.9.3	Activating safety mode	101
4.9.4	Downloading the safety program	102
4.10	Working with safety-related parameters on a PCS 7 OS.....	104
4.10.1	Safety Data Write (SDW)	104
4.10.2	F_QUITES	106
4.11	Creating F block types	107
4.11.1	Rules for F-block types	107
4.11.2	Creating F block types using "Compile chart as F block type"	108

4.12	Monitoring times and system response times.....	110
4.12.1	Calculating the F-cycle monitoring time (for block F_CYC_CO).....	111
4.12.2	Communication monitoring time for F CPU/F signal modules.....	116
4.12.3	Monitoring time for safety-related communication between F-CPU's	121
4.12.4	Monitoring communication between F-shutdown groups	123
4.12.5	Response times of safety functions	123
5	Configuration with Safety Matrix	129
5.1	Creating and configuring a Safety Matrix.....	129
5.2	Documenting a Safety Matrix.....	129
5.3	Organizing matrices into different shutdown groups.....	129
5.4	Duplicating matrices.....	134
5.5	User authorizations for the Safety Matrix Viewer.....	135
5.6	Interconnections between the matrix and safety program.....	138
5.7	Using imported F channel drivers in the matrix	138
6	Changes, tracking changes, and acceptance	139
6.1	General information	139
6.2	Preparing for changes.....	140
6.3	Changes in CFC	142
6.4	Changes in HW Config	143
6.5	Downloading changes/Complete downloading.....	143
6.6	Tracking changes in the safety program.....	144
6.7	Printing program data	146
6.8	Plant acceptance	148

Installing the fail-safe system

1.1 Software components

This documentation is based on PCS 7 versions V6.1 SP2 and V7.0 SP1, as well as the following software components:

- S7 F Systems
 - S7 F Configuration Pack V5.5 SP3
 - S7 F Systems V6.0
 - S7 F Systems Lib V1_3 and Lib V1_2
 - S7 F Systems HMI V5.2 SP3
- The following components are also required for using the F-AI 6 x 0/4...20mA HART module:
 - S7 F Configuration Pack V5.5 SP5
(<http://support.automation.siemens.com/WW/view/en/15208817>)
 - Hardware Upgrade Package (HUP) for using the 6 F-AI HART module (without HART function) in SIMATIC PCS 7 V6.1 SP2 and V7.0 SP1 (EU)
"PCS7_V61SP2-70SP1_Package_F-AI6-V1.exe"
(<http://support.automation.siemens.com/WW/view/en/31481983>)
- or
 - Hardware Upgrade Package (HUP) for using the F-AI HART, 6ES7 336-4GE00-0AB0, in SIMATIC PCS 7 V7.0 SP1 (EU)
"PCS7_V70_SP1_HUP_F-AI6_HART_V1.exe"
(<http://support.automation.siemens.com/WW/view/en/29000518>)
- Safety Matrix Engineering Tool
 - Safety Matrix Engineering Tool V6.1
 - Safety Matrix Viewer V6.1
 - Safety Matrix AS-OS Engineering V6.1

A "S7 F Systems RT License" is also required for operating a safety-related automation system.

All SIMATIC software must be closed during the installation process.

Note

For more information, please refer to the *"S7 F/FH Systems Configuring and Programming"* Manual (<http://support.automation.siemens.com/WW/view/en/2201072>)

or the *"Industrial Software Safety Matrix"* Manual.
(<http://support.automation.siemens.com/WW/view/en/19056619>)

1.2 Installation on the PCS 7 engineering station (ES)

1.2.1 Installing S7 F Systems

Procedure

Run SETUP.EXE to start the installation and follow the instructions in the setup program.

The following components must be selected for installation:

- S7 F Configuration Pack V5.5 SP3
- S7 F Systems V6.0
- S7 F Systems Lib V1_3

You must also install the following components in order to use the F-AI 6 x 0/4..20mA HART module:

- S7 F Configuration Pack V5.5 SP5
- HUP: PCS7_V61SP2-70SP1_Package_F-AI6-V1.exe for operation without the HART function
or
- HUP: PCS7_V70_SP1_HUP_F-AI6_HART_V1.exe for operation with the HART function

The links for downloading this additional software can be found in Section Software components (Page 9).

Note

S7 F Systems V6.0 can also be used with S7 F Systems Lib V1_2, which you will find in the "Options" directory of the S7 F Systems Product CD.

If you are using Safety Data Write (SDW), you will also need to select the following option:

- S7 F Systems HMI V5.2 SP3

1.2.2 Installing Safety Matrix

Procedure

Run SETUP.EXE to start the installation of the Safety Matrix; follow the instructions in the setup program and select the following components:

- Safety Matrix Engineering Tool V6.1
- Safety Matrix Viewer V6.1
- Safety Matrix AS-OS Engineering V6.1

1.3 Installation on the PCS 7 operator station (OS)

The next section outlines the installation steps for an OS client. Please follow these if you are using an OS single station or an OS server for operation.

1.4 OS client installation

1.4.1 Installing S7 F Systems

Procedure

If you are using SDW, run SETUP.EXE to start the installation and install the following option:

- S7 F Systems HMI V5.2 SP3

1.4.2 Installing Safety Matrix

Procedure

Run SETUP.EXE to start the installation. Follow the instructions in the setup program and select the following component:

- Safety Matrix Viewer V6.1

Advanced PCS 7 ES settings

2.1 Access protection

An S7-400F/FH system being operated as a safety system is protected by two passwords:

- The CPU password is configured in the hardware configuration and is intended to protect the CPU against accidental downloading or the wrong program being downloaded.
- Another password is configured in the "Edit Safety Program" dialog box, and provides the fail-safe user program with protection against compilation. Changes made to the fail-safe user program can only be compiled if this password has been activated.

Note

In the case of PCS 7 V7.0 and higher with SIMATIC Logon, it is also possible to set a project password for protecting a PCS 7 project.

Access to the ES is also protected by a password at operating-system level (Windows login, screen saver) in most cases.

2.1.1 General measures

You can restrict access to the ES by implementing access protection for authorized persons: To do this you can, for example, install the ES in an operating area that can only be accessed by these persons.

You can use the options offered by the operating system by protecting the ES with a password and setting a screen saver that is activated automatically.

To protect the safety program, you can implement organizational measures to ensure that:

- The passwords for the CPU and safety program can only be accessed by authorized persons.
- Authorized persons expressly reset access authorization for the F CPU before exiting the ES (by selecting "CPU > Access authorization > Cancel" or closing all applications in the SIMATIC Manager).

2.1.2 Access protection with SIMATIC Logon

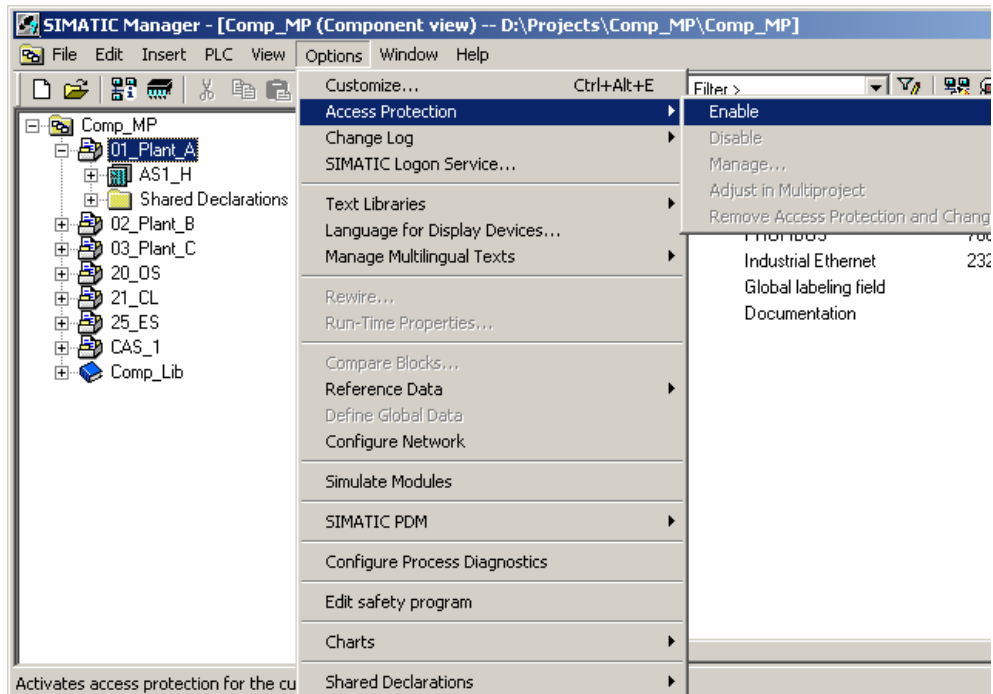
With PCS 7 V7.0 and higher, it is possible to set up access protection for individual subprojects with SIMATIC Logon. With a station-selective multiproject structure, this means that it is possible to assign access rights for protecting AS projects with F program.

Requirement

The SIMATIC Logon Service V1.3 SP1 or higher is installed.

Procedure

1. In the SIMATIC Manager, activate access protection on a selected project node via "Options > Access Protection".



The project format is changed the first time access protection is activated. A message appears indicating that the modified project can no longer be edited with older versions of STEP 7 (< 5.4).

2. This is followed by logon to the SIMATIC Logon Service.

The Windows user activating access protection is entered automatically as the first project administrator. The project password is set at this time.

Note

When a multiproject is opened without prior authentication, the projects with activated access protection are displayed grayed-out.

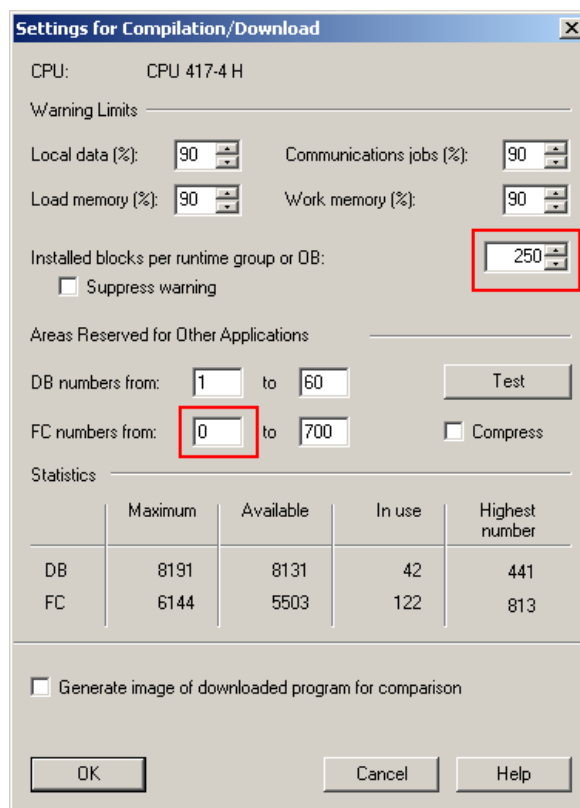
2.2 CFC settings for compiling and downloading

Procedure

1. For compiling programs with F blocks, set the threshold for generating the warning based on the number of blocks per runtime group. The default setting is 50. We recommend changing the setting to 250 for F programs.

Note

This value applies for any project that is compiled on the ES and should be reset for the purpose of compiling standard programs.



2. In the "Areas Reserved for Other Applications" field, set the value for "FC numbers from:" to 0. The default setting is 60.

Note

Please note that blocks FC 0 and SFC 0 are not called in your program. For more information, please refer to the following FAQ on the Service & Support pages:

"What should you watch out for when using F block types with S7 F Systems V 6.0 and FC 0 or SFC 0 in the project?"

<http://support.automation.siemens.com/WW/view/en/32202151>

Configuring S7 F/FH hardware

The following CPU types have been released for using S7 F systems in conjunction with PCS 7:

- CPU 412H
- CPU 414H
- CPU 417H

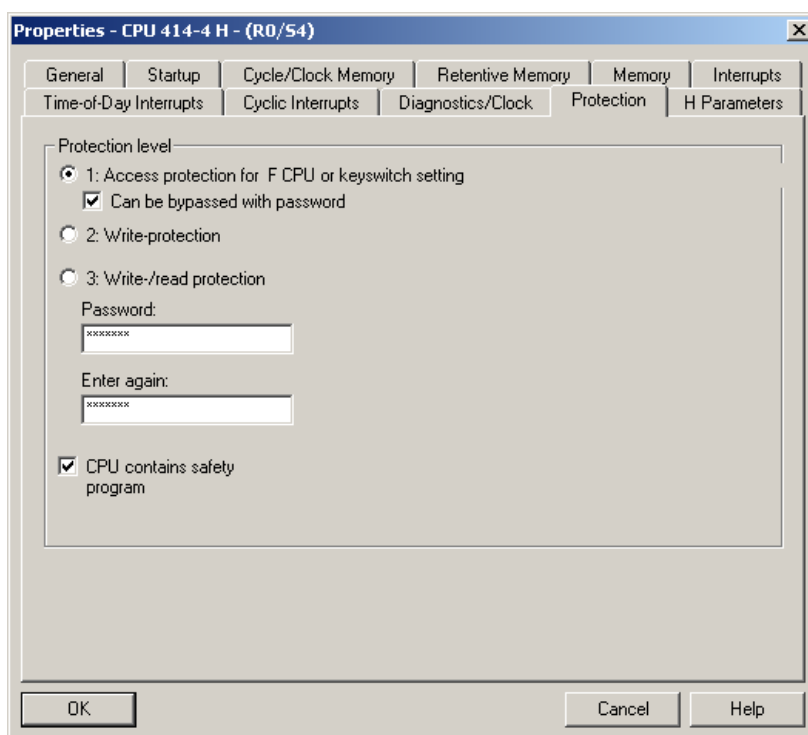
In the SIMATIC PCS 7 catalog, safety-related automation systems can be configured and ordered as bundles with a single or redundancy station in various designs.

3.1 Adapting CPU parameters (single F-system)

3.1.1 Password and access protection

In order to activate the safety functions contained in the H-CPU's operating system, you need to enter a password. A prompt appears accordingly on CPU download.

The "CPU contains safety program" option also needs to be activated.



Note

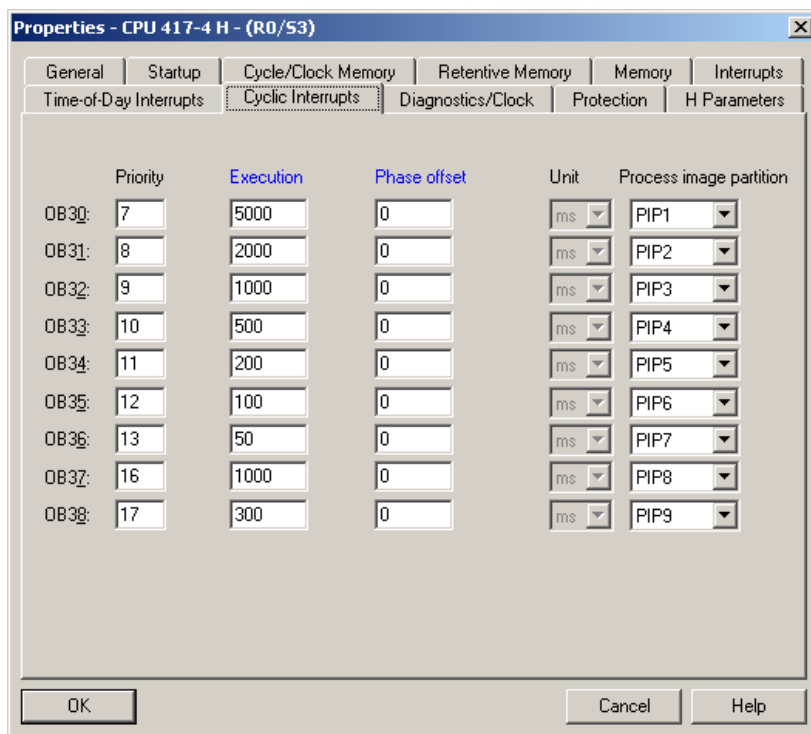
Protection level 1 needs to be configured so that the prompt to enter the F CPU password does not appear in the event of changes to the standard user program.

A password is also assigned to the fail-safe program; this is created when the user program is generated. This password must not be the same as the CPU password.

3.1.2 Cyclic interrupts

Parameter assignment

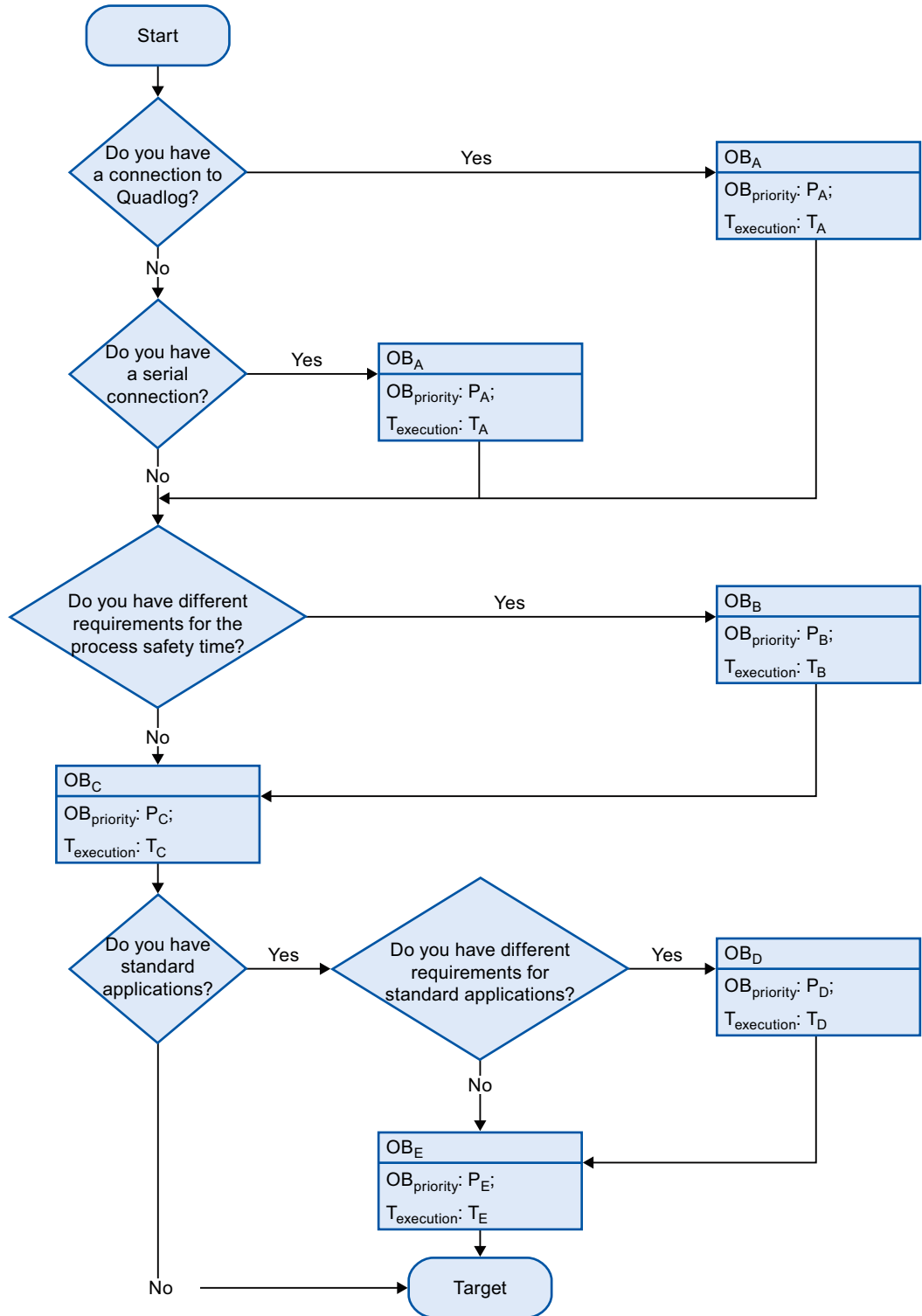
Process image partitions do not need to be configured for F program parts. In S7 F/FH systems, F module drivers are used instead of the process image to access the F signal modules. A process image is required, however, if standard program parts are also to be used in the cyclic interrupt OB.



Note

No reduction ratio or phase offset may be configured for cyclic interrupt OBs with F program.

The flow diagram below demonstrates one method you can use for structuring your program according to the requirements of your process, using the cyclic interrupt OBs.



The user program is divided to cyclic interrupt OBs according to the requirements:

- OB_A
 - Very fast applications
 - for example connections, such as Quadlog or Modbus
- OB_B
 - Fast F applications (cyclic interrupt with special treatment)
 - For example pressure protection functions
- OB_C
 - (Slow) F applications
 - If there is only one cyclic interrupt for F applications, then it will be the cyclic interrupt OB with special treatment.
 - For example slow temperature protection functions
- OB_D
 - Fast standard applications
 - For example fast control functions
- OB_E
 - Slow standard applications
 - For example temperature measurements, visualization functions, enable functions

The following rules apply:

- OB_A to OB_E:
 - One of the cyclic interrupt OBs from OB30 to OB38
 - Only certain cyclic interrupt OBs are available, depending on the CPU type.
- T_{execution}: T_A < T_B < T_C < T_D < T_E
(T_{execution} = The execution time of the cyclic interrupt OBs in ms configured in the CPU properties)
- OB_{priority}: P_A > P_B > P_C > P_D > P_E
(OB_{priority} = The priority of the cyclic interrupt OBs configured in the CPU properties. Possible values are 7 to 18.)

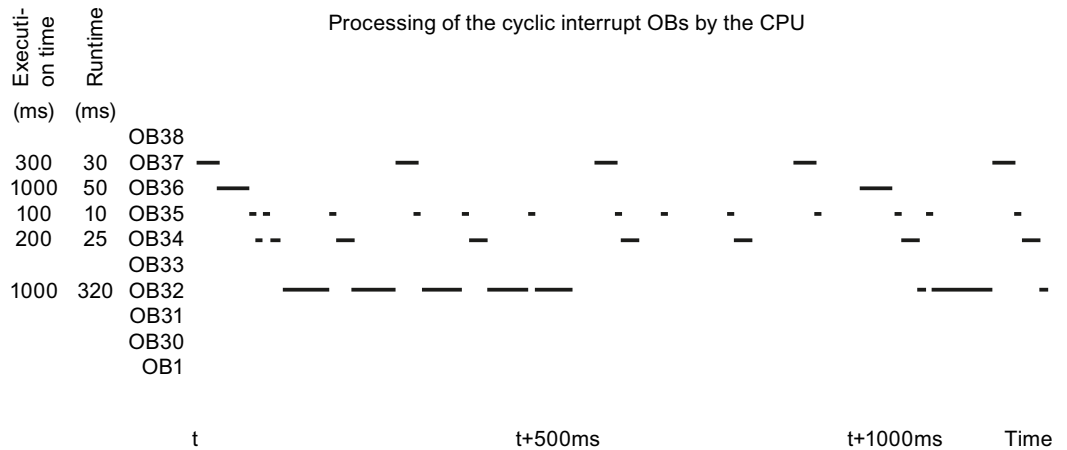
Example 1: Reconfiguration (no Quadlog connection, no serial communication)

OB	Prio	Call interval	Purpose	Comment
OB 38	15	10 ms	Empty	
OB 37	17	300 ms	Fast F application	OB with special handling
OB 36	16	1,000 ms	F application	
OB 35	12	100 ms	Fast standard application	Ex.: Control
OB 34	11	200 ms	Standard application	Ex.: Pressure measurement
OB 33	10	500 ms	Empty	
OB 32	9	1,000 ms	Slow standard application	Ex.: Temperature measurement, visualization function, enable function
OB 31	8	2,000 ms	Empty	
OB 30	7	5,000 ms	Empty	

Note

If you are using time stamping of the IM module, the setting for OB 40 should be changed from Prio16 to Prio18.

The figure below shows how the individual cyclic interrupt OBs from Example 1 are processed over a period of 1,280 ms, starting from the point in time (t) at which the CPU begins program processing. In this example, the CPU utilization is at approximately 65%.



Example 2: Reconfiguration/migration, for which the hardware present must be connected via serial communication (Modbus)

OB	Prio	Call interval	Purpose	Comment
OB 38	18	50 ms	Serial communication	Ex.: Modbus
OB 37	17	300 ms	Fast F application	OB with special handling
OB 36	16	1,000 ms	F application	
OB 35	12	100 ms	Fast standard application	Ex.: Control
OB 34	11	200 ms	Standard application	Ex.: Pressure measurement
OB 33	10	500 ms		
OB 32	9	1,000 ms	Slow standard application	Ex.: Temperature measurement, visualization function, enable function
OB 31	8	2,000 ms	Empty	
OB 30	7	5,000 ms	Empty	

Note

If you are using time stamping, the setting for OB 40 should be changed from Prio16 to Prio19.

Example 3: Reconfiguration/migration, for which the Quadlog hardware present must be connected via DP/IO bus link

OB	Prio	Call interval	Purpose	Comment
OB 38	17	10 ms	Quadlog connection	
OB 37	16	300 ms	F application	OB with special handling
OB 36	13	50 ms	Empty	
OB 35	12	100 ms	Empty	
OB 34	11	200 ms	Empty	
OB 33	10	500 ms	Empty	
OB 32	9	1,000 ms	Slow standard application	Ex.: Temperature measurement, visualization function, enable function
OB 31	8	2,000 ms	Empty	
OB 30	7	5,000 ms	Empty	

Note

If you are using time stamping of the IM module, the setting for OB40 should be changed from Prio16 to Prio18.

CPU utilization

During configuration, it must be ensured that the sum of the processing times for the individual cyclic interrupt OBs does not exceed 70% of the maximum CPU load for a fault-tolerant S7-400 system and 80% for a standard S7-400 system in stand-alone operation.

Note

To help you determine the processing times of the individual cyclic interrupt OBs, please refer to the FAQ *"How can you calculate the cycle load of the automation system (AS) online?"* (<http://support.automation.siemens.com/WW/view/en/22000962>)

As of PCS 7 V7.0 SP1 and an S7-400H CPU with FW 4.5 and higher, the runtimes for the cyclic interrupt OBs may also be read at the CPU_RT block.

As of PCS7 V7.0 with PCS 7 Library V7.0 the block CPU_RT will be implemented in each CPU. This block prevents a CPU STOP as the result of a cycle overload. thanks to a two-stage "load-shedding" process:

- In the first stage of load shedding, all cyclic interrupt OBs that have not been excluded by the project engineer are not processed for one cycle.
- In the second stage, none of the cyclic interrupt OBs are processed for one cycle (even the ones that were excluded previously). If this does not have a steadying effect, whenever cyclic interrupt OBs are processed their processing will be suspended again for one cycle.

Note

In order to prevent F cycle-time monitoring in the CPU and the F monitoring time on the F signal modules from being triggered, and the system reaction time from being extended, you must exclude the cyclic interrupt OBs with F program during the first stop prevention stage.

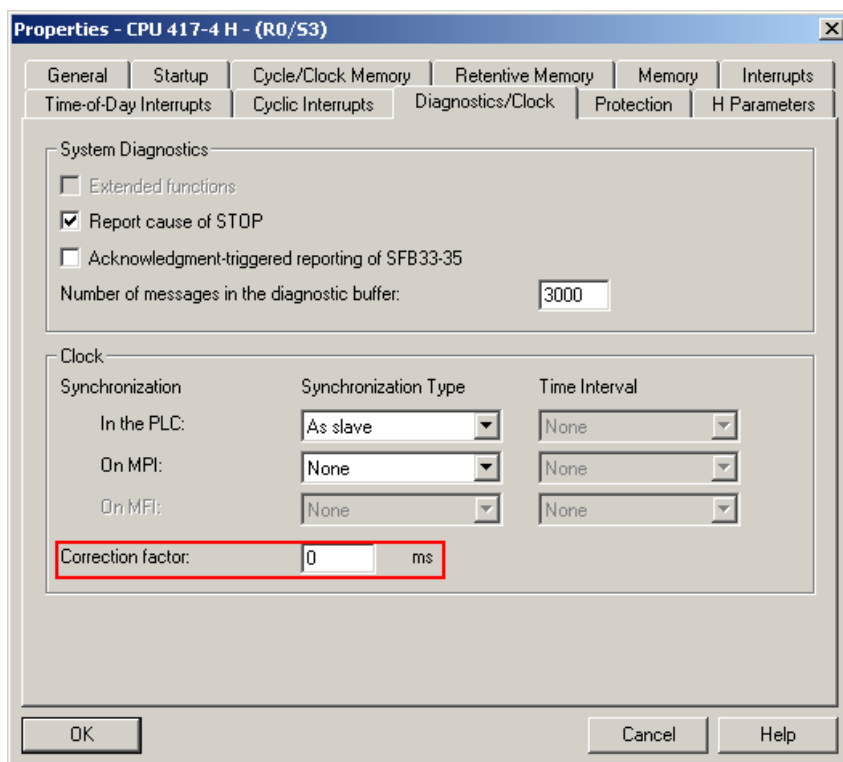
Alternatively, you can disable stop prevention altogether by setting the parameter for input Max_RTRG of block CPU_RT to 0. You can find this block in the @CPU_RT chart.

For more information on the subject of CPU stop prevention, please refer to the help for the CPU_RT block.

3.1.3 Diagnostics/Clock

For process data to be compatible for evaluation, all components of the process control system must work with the same time of day so that messages – regardless of the time zone in which they are generated – can be assigned correctly in terms of temporal sequence. This usually involves an OS server or an external time-of-day master (SICLOCK) taking on the role of the time-of-day master. All other operator stations and automation systems on the plant bus then have the time from this time master and, therefore, identical time.

This is why the synchronization in the AS is always set to the "As slave" synchronization type for each AS/CPU in a time-synchronized PCS 7 plant.



Check that the correction factor is set to 0 ms under "Time" on the "Diagnostics/Clock" tab in a CPU with safety program.

Note

See also Compendium Part A "Configuration of the Hardware > General CPU Settings > Diagnostics/Clock".

3.1.4 H parameters

Self-test (advanced CPU test)

The CPU tests the following during the self-test:

- Processor
- Internal memory
- I/O bus

If the test detects any faults, they are reported and the CPU goes to STOP.

Test cycle time

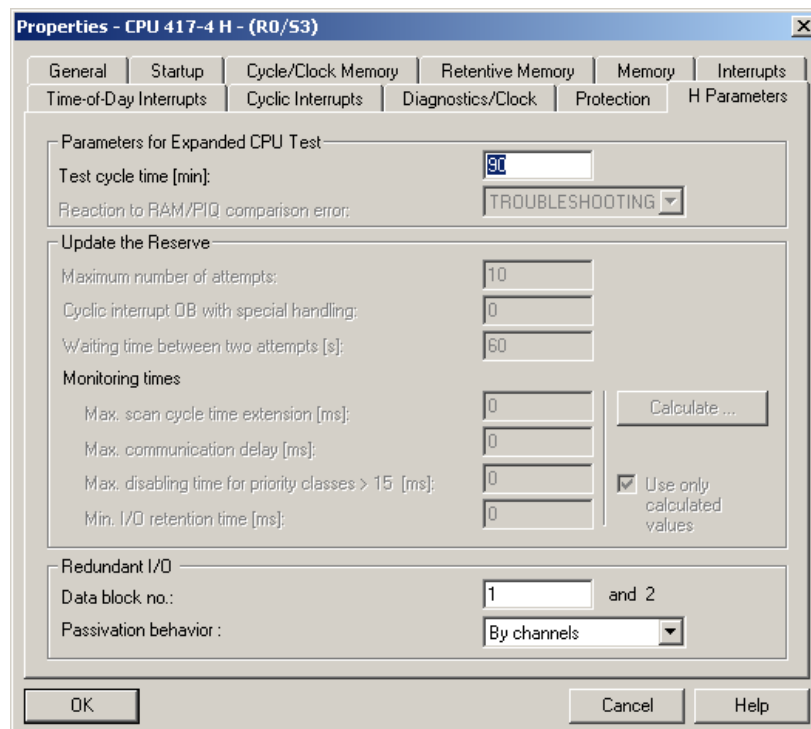
The test cycle time (default setting: 90 minutes) indicates the time taken for a complete background self-test.

Note

For S7 F/FH systems, this parameter can be increased up to a maximum of 12 hours (720 minutes).

In S7 F/FH systems, the test may not be modified by calling SFC 90, "H_CTRL". Otherwise, the safety program will go to F-STOP after a maximum of 24 hours. Switching test components on or off is prohibited.

In the event of a STOP, the self-test enters the following diagnostics event in the diagnostic buffer of the F CPU: "Safety program: Error detected" (event ID 16#75E1).



Note

If you are using redundant standard signal modules, the system will generate two data blocks in the AS: DB 1, with the number specified, and DB 2, with the number following this.

Make sure that these blocks are not used twice (e.g. in a Modbus connection). If necessary, adjust the DB numbers in the H parameters.

3.2 Adapting CPU parameters (fault-tolerant F system)

All settings made in the single F system must also be made in the fault-tolerant system.

Note

Parameters in blue can be changed during operation on an H system.

The following settings must also be made when using redundant CPUs:

- Reaction to RAM/PIQ comparison error
- Cyclic interrupt OB with special handling
- Monitoring times

3.2.1 Reaction to RAM/PIQ comparison error

Troubleshooting

ERROR-SEARCH mode is set by default in response to a comparison error (default response). The purpose of ERROR-SEARCH mode is to detect and identify a faulty CPU.

Select how the H system should react to an error generated during the comparison of the RAM areas and the process images of the outputs:

- Troubleshooting:

The TROUBLESHOOTING operating state is only available in the redundant system mode.

During the self-test, the master and standby CPUs are compared. If the test reveals any differences between them, an error will be reported. Possible types of errors are hardware errors, checksum errors, and RAM/PIQ comparison errors.

Note

To find out which events trigger the TROUBLESHOOTING operating state, please refer to the "TROUBLESHOOTING operating state" section in the manual titled *"Fault-tolerant Systems S7-400H"*.

- H system STOP:
The entire H system is set to the STOP system mode.
- Standby CPU STOP:
The standby CPU is set to the STOP mode. The master CPU remains in RUN (solo system mode).

NOTICE

Processing is also paused briefly in the master CPU for the purpose of updating a CPU that is starting up (see Monitoring times).

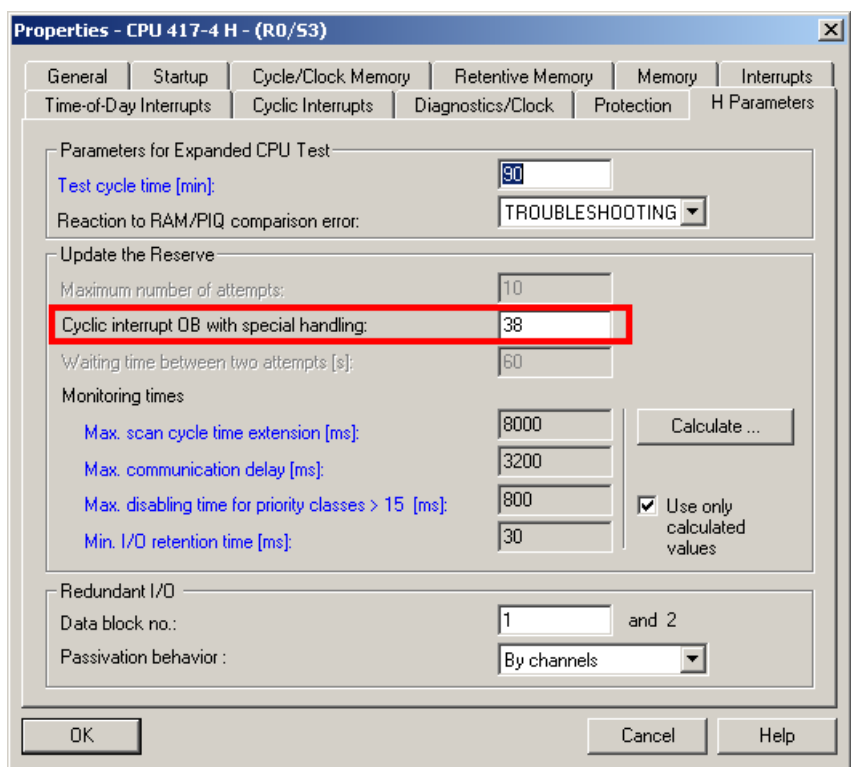
If this is only to occur at certain times defined by the plant operator, "Standby STEO" must be selected as the error reaction. Following this, maintenance personnel must carry out a self-test on both CPUs at an appropriate point in time, by performing a non-buffered CPU startup.

3.2.2 Cyclic interrupt OB with special handling

Priority class

In order to prevent time monitoring (F-CYC_CO or monitoring time of modules) from being triggered in the event of a CPU that is starting up in the H system being coupled/updated, you need to set the priority of the cyclic interrupt OBs allocated to the F program (OB 30 to OB 38) to > 15 on the "Cyclic Interrupts" tab (see Monitoring times).

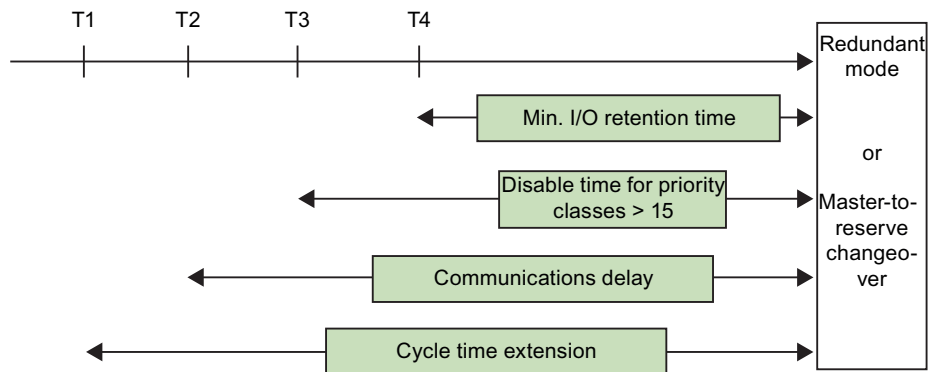
"Cyclic interrupt OB with special handling" is an H parameter containing the number of the cyclic interrupt OB which is called specifically by the operating system when the standby CPU is updated, once all interrupts have been disabled. Here, enter the number of the cyclic interrupt OB with the highest priority, to which F blocks are assigned in CFC.



3.2.3 Monitoring times

While the standby CPU is being updated, the FH system checks that the scan cycle time extension, the communication delay, and the disabling time for priority classes > 15 do not exceed the maximum values you have set. It also ensures that the minimum I/O retention time that has been configured is observed.

The times of relevance to the update process are summarized in the figure below.



T1: End of active OBs up to priority class 15

T2: Stopping of all communication functions

T3: End of cyclic interrupt OB with special handling

T4: End of copying of outputs to reserve CPU

If the update fails due to a maximum value being exceeded, the CPU will continue to run in solo mode and try again to update the reserve CPU once the specified wait time has elapsed.

Note

You can find more information in the manual titled *"S7-400H Automation Systems, Fault-Tolerant Systems"*.

If the "Use only calculated values" box is checked (recommended default setting), it will not be possible to enter or modify the monitoring times manually. The best values for the user program can then be determined automatically by clicking the "Calculate..." button.

Note

We recommend only using calculated values.

3.2.4 Calculating monitoring times

Calculation

You can use this dialog to calculate monitoring times for updating the standby CPU.

You need to enter information about your user program for this purpose:

- Runtime of cyclic interrupt OB with special handling
- Work memory used for all data blocks in the user program

Defaults from the process (monitoring times) and the current configuration (bus parameters, number and type of DP slaves, etc.) are also used in the calculation.

Section	Parameter	Value
User program information	Time interval of high priority watchdog interrupt or watchdog interrupt with special handling [ms]	300
	Runtime of the watchdog interrupt concerned [ms]	50
	Work memory used for all data blocks in the user program [Kbyte]	1500
Process data (safety times)	Most critical F-SM monitoring time [ms]	2500
	In DP master system:	1.2
Calculated monitoring times	Maximum scan cycle time extension [ms]	11300
	Maximum communication delay [ms]	4520
	Maximum disabling time for priority classes > 15 [ms]	1130
	Minimum I/O retention time [ms]	30

Run time of the cyclic interrupt concerned

If a cyclic interrupt OB with special handling has already been configured, half its execution interval will be entered as the default setting for "Runtime of the watchdog interrupt concerned".

Here, enter the actual runtime of the cyclic interrupt with special handling, plus some reserve time (10 to 20%).

Note

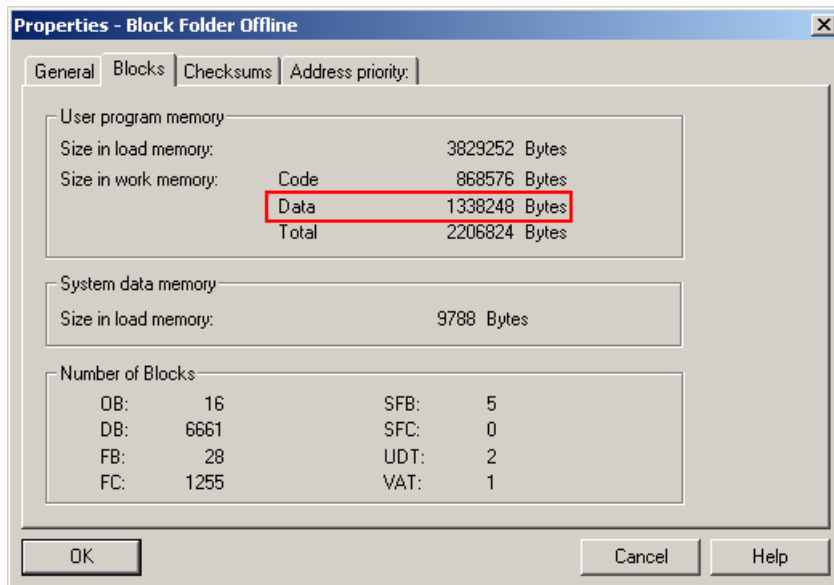
This runtime can be calculated using the TIME_BEG and TIME_END blocks. See the following FAQ: *"Determining the processing time of cyclic interrupt OBs"*

With PCS 7 V7.0 and higher, and an S7-400H CPU with FW 4.5 and higher, the runtimes for the cyclic interrupt OBs may also be read at the CPU_RT block.

Work memory allocation (data memory)

The work memory allocation comprises all data blocks; in other words, it also includes DBs generated dynamically. A value of 1,024 KB is entered here by default. This value should be modified to reflect the actual data memory requirements of the user program. We recommend adding an expansion reserve of approximately 10%.

The allocation of work memory for data blocks can be read in SIMATIC Manager by selecting the block container with the menu command "Edit > Object Properties".



Calculation

Once all the parameters have been set, the values are calculated by pressing the "Recalculate" button.

If the F signal modules are configured in safety mode in HW Config, the "Max. disabling time for priority classes > 15" parameter is calculated using the following formula:

- T_{P15} (DP master system) = $T_{PTO} - (2 \times T_{TR} + T_{CI} + T_{PROG} + T_{DP_ST} + T_{SLAVE_ST})$
(determined for each DP master system)
- T_{P15_UP} in ms = $0.7 \times \text{size of DBs in work memory in bytes} / 1,024 + 75$

where:

- T_{PTO} = Shortest F monitoring time configured on an F signal module
- T_{TR} = Target rotation time of the PROFIBUS line
- T_{CI} = Call time of the cyclic interrupt OB
- T_{PROG} = Runtime of the cyclic interrupt OB
- T_{DP_ST} = DP switchover time
- T_{SLAVE_ST} = Slave switchover time
- T_{P15_UP} = Time for copying the DBs of the user program (UP)

Calculating the parameters may cause one of the following messages to appear:

- "The F monitoring times cannot be adhered to due to the configuration specified. Times which can no longer be used were output."

if:

"Maximum disabling time for priority classes > 15" < "Minimum I/O retention time" + 50 ms

- "Due to the configuration specified, monitoring times have been calculated to ensure that the F monitoring time can be adhered to. It cannot be guaranteed that each attempt at coupling and updating will be successful."

if:

T_{P15} (DP master system) < T_{P15_UP}

The "Maximum communication delay" and "Maximum scan cycle time extension" are calculated from the "Maximum disabling time for priority classes > 15" by multiplying this setting once by 4 and once by 10.

Note

When determining the shortest F monitoring time configured on an F signal module, none of the monitoring times set in PROFIsafe PA devices are taken into account.

Maximum disabling time for priority classes

Check the value for "Maximum disabling time for priority classes > 15".

Note

We recommend doing this again on the plant once commissioning is complete.

Follow the steps outlined below:

- Start up a CPU in an H system.
- In the diagnostic buffer of the master CPU, search for messages relating to the master CPU's "Transition from coupling to updating", as well as for the message "Redundant mode".
- The "Maximum disabling time for priority classes > 15" must be greater than the time difference between the two messages. Make sure you factor reserve time of approximately 20% into your setting.

Note

You can find more information and corrective measures in the manual titled *"Fault-tolerant Systems S7-400H"*.

See also

<http://support.automation.siemens.com/WW/view/en/1023077>

3.3 Communications module parameters/Networks

The settings for communication modules are explained in Compendium Part A.

It may be advisable to operate the F I/O on a separate DP master system when there are numerous nodes or nodes with low transmission speeds.

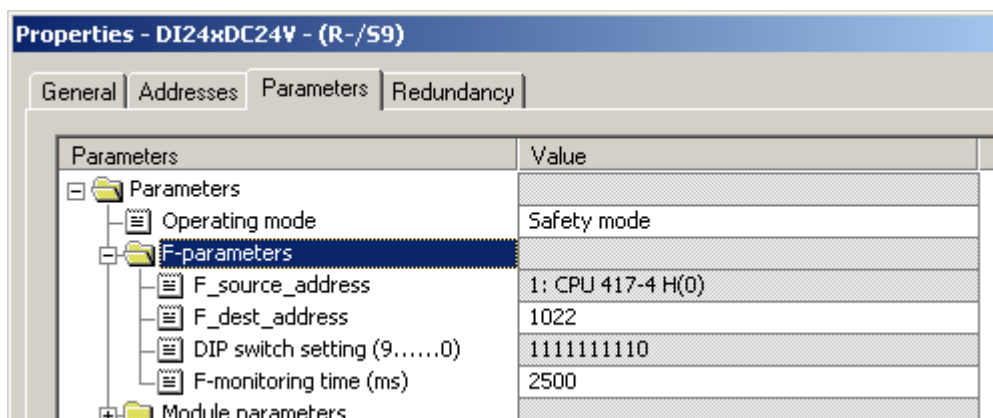
3.4 Setting system parameters for F-signal modules

Similar to standard modules, F signal modules are configured in HW Config. This requires the corresponding F Configuration Pack.

Unused channels can be added during operation, provided that, during first commissioning, they have been activated in HW Config and equipped with resistors in order to suppress channel faults.

Procedure

Once you have added the F signal modules to the ET 200M station in HW Config, you can access the configuration dialog by selecting the menu command "Edit > Object Properties" or double-clicking the corresponding F signal module.



3.4.1 Operating mode

Safety mode

"Safety mode" must be selected for the signal modules (there may be a difference between SIL 2 and SIL 3).

3.4.2 PROFIsafe addresses

F source and destination addresses

The PROFIsafe addresses (F_SOURCE_ADD, F_DEST_ADD) are used to uniquely identify the source and destination during PROFIsafe communication. The F_DEST_ADD uniquely identifies the PROFIsafe destination (the module). The F_DEST_ADD must, therefore, be unique across both the network and the station. The F_SOURCE_ADD is permanently assigned to the CPU.

To prevent parameter assignment errors, the F_SOURCE_ADD and the F_DEST_ADD are assigned automatically.

DIP switch settings

The DIP switch setting is the binary representation of the F_DEST_ADD. It must be set on the F signal module DIP switches before you install the F signal module.

Standard mode	Safety mode
<p>ON</p> <p>4096 2048 1024 512 256 128 64 32 16 8</p> <p>or</p> <p>ON</p> <p>4096 2048 1024 512 256 128 64 32 16 8</p>	<p>All possible combinations that do not correspond to standard mode</p> <p>ON</p> <p>4096 2048 1024 512 256 128 64 32 16 8</p> <p>9 8 7 6 5 4 3 2 1 0</p> <p>Module start address 192 corresponds to F_DEST_ADD 24</p>

When these modules are in standard mode, F_DEST_ADD is always set to "0" (delivery condition).

In the case of older module versions, the F_DEST_ADD is dependent on the module start address. The following applies: $F_DEST_ADD = \text{Module start address} / 8$.

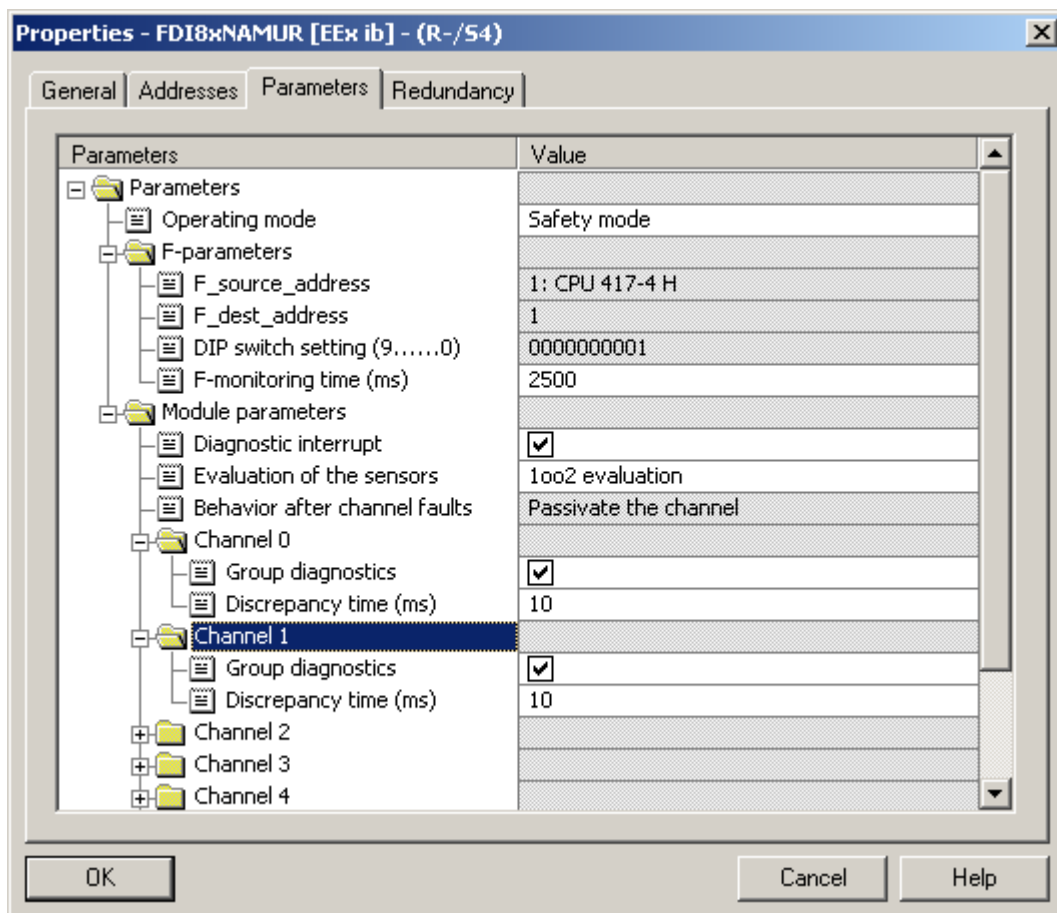
F monitoring time

If you are operating the F signal module in safety mode, this is where you set the F monitoring time for safety-related communication between the F CPU and F signal module (PROFIsafe monitoring time).

Note

You can find information about setting the PROFIsafe monitoring time in the section titled Monitoring times and system response times (Page 110).

3.4.3 Module parameters - general



Diagnostic interrupt

To enable the PCS 7 driver blocks to report interrupts, the diagnostic interrupt for the F signal module must always be activated in safety mode.

Various error events, which the fail-safe signal module can define using its diagnostics function, trigger a diagnostics interrupt. The diagnostics events which occur are made available by the F CPU module.

Group diagnostics

If you check this box for a specific channel, a channel-specific event (a wire break, for example) will trigger an error reaction in the safety program (the substitute value is activated on the the channel driver and QBAD is set). If "Enable diagnostics interrupt" is selected, a diagnostics interrupt will be triggered in the CPU and a corresponding process control message will be sent on the OS.

The "Group diagnostics" parameter is used to activate and deactivate the transfer of channel-specific diagnostic messages (e.g. wire break, short circuit) on F signal modules to the CPU.

For the following F signal modules, group diagnostics needs to be activated whenever a channel is activated:

- SM 326; DI 24 x DC 24 V (order no. 6ES7326-1BK00-0AB0)
- SM 326; DI 8 x NAMUR (order no. 6ES7326-1RF00-0AB0)
- SM 336; AI 6 x 13 Bit (order no. 6ES7336-1HE00-0AB0)
- SM 326; DO 10 x DC 24V/2A (order no. 6ES7326-2BF00-0AB0)

With all other F signal modules, this takes place automatically when you activate a channel.

To maintain an overview, you should deactivate group diagnostics on input or output channels which are not in use on the F signal modules or wire the module so that no channel errors occur.

Note

Where fail-safe input and output modules in safety mode are concerned, group diagnostics must be active on all connected channels. Please check that group diagnostics has only been deactivated for input and output channels which are not in use.

3.4.4 Activating channels

Due to the structure of fail-safe signal modules, it is not possible to make changes to their hardware configuration or to download them without the module being passivated. Passivating output modules involves establishing a safe state on all outputs, while passivating input modules involves the input drivers outputting the value 0.

To enable you to activate free channels on F signal modules during operation, you must activate the channels in HW Config. Due to the diagnostics for the F signal modules, however, activated channels lead to pending errors, which you can suppress by equipping the channels with equivalent resistors.

SM326; DI 8 x NAMUR [EEx ib] (6ES7 326-1RF00-0AB0)

- Determine module parameters
- Activate group diagnostics for the channel
- Connect the channel I/Os to a resistor (e.g. 1 kohm)

SM326; DI 24 x DC 24V (6ES7 326-1BK01-0AB0)

- Determine module parameters
- Determine sensor supply
- Activate channel
- Determine sensor evaluation
- Determine type of sensor interconnection
- If necessary, set discrepancy reaction and discrepancy time
- Provided that the sensor supply comes from the module:
Connect input to the sensor supply via a resistor (e.g. 1 kohm)

SM326; DO 10 x DC 24V/2A (6ES7 326-2BF01-0AB0)

- Determine module parameters
- Activate group diagnostics for the channel
- To simulate an actuator, interconnect output with a resistor (e.g. 2.7 kohms) downstream of the ground connection

SM326; DO 8 x DC 24V/2A PM (6ES7 326-2BF40-0AB0)

- Determine module parameters
- Activate channel
- Set diagnostic wire break
- If function is activated:
Connect both channel I/Os to a resistor (e.g. 2.7 kohms)

SM 336; F-AI 6 x 13 Bit (6ES7 336-1HE00-0AB0)

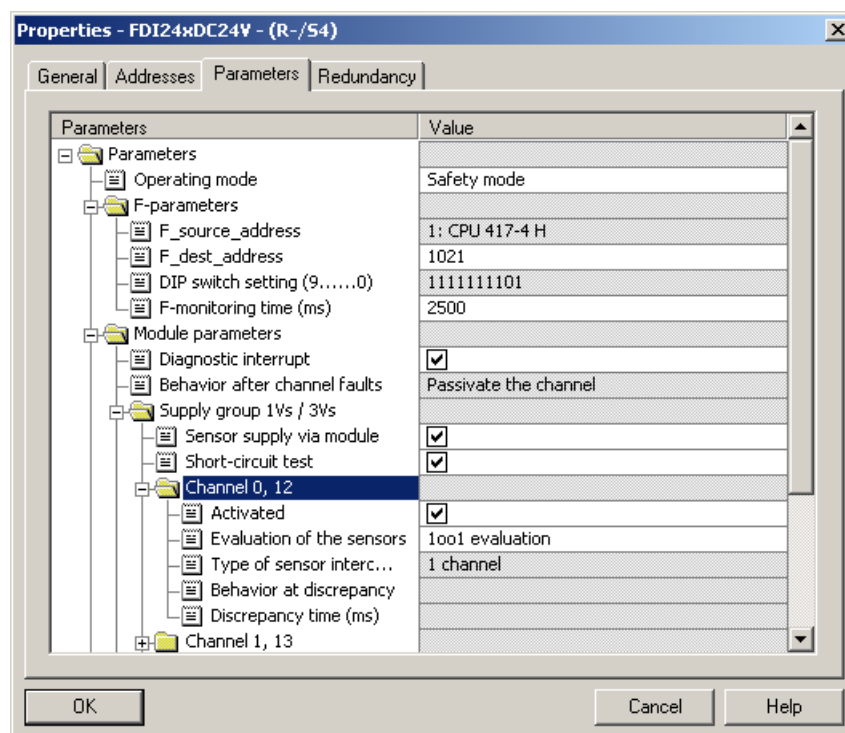
- Determine module parameters
- Activate group diagnostics for the channel
- Make channel-specific settings
- Interconnect plus input of channel with supply voltage via a resistor (e.g. 3.9 kohms) and connect minus input to ground

SM 336; F-AI 6 x 0/4...20mA HART (6ES7 336-4GE00 0AB0)

- Determine module parameters
- Determine type of sensor interconnection
- Make channel-specific settings
- Make HART communication settings
- Interconnect plus input of channel with supply voltage via a resistor (e.g. 3.9 kohms) and connect minus input to ground

Note

You can find more information on activating channels during operation in the following FAQ: *"Programming with F/FH systems - Changing parameters on fail-safe I/Os"* (<http://support.automation.siemens.com/WW/view/en/21382997>)

3.4.5 Assigning parameters for the SM326; DI 8 x NAMUR/SM326; DI 24 x DC 24V**Sensor supply via module**

You can set whether the sensor is supplied via the F signal module here. If it is, you can also activate a short-circuit test for this supply.

Short-circuit test

You can use this parameter to activate short-circuit detection for the F signal module.

The short-circuit test can only be activated for sensors that are supplied by the F signal module.

Short-circuit detection disconnects the sensor supply briefly and tests the input signal. A cross circuit is detected between the channels and an L+ fault at the active inputs.

Whenever a short circuit is detected, the F signal module will trigger a diagnostic interrupt on the CPU and send a corresponding process control message to the OS.

Sensor evaluation

- 1oo1 (1v1) evaluation

A sensor connected to the F signal module via a single channel

- 1oo2 (2v2) evaluation

For a process signal one or two sensors are connected to two opposite inputs on a F signal module. The signal states of inputs (equivalence or non-equivalence) are compared internally.

The following safety classes can be achieved:

- 1-channel – SIL 2; in the case of multiple channels SIL 3 can be achieved by means of voting in the CPU.
- 2-channel – SIL 3 (voting on module)

Type of sensor interconnection

If "1oo2 sensor evaluation" is selected, you can select the type of sensor interconnection for each input channel here (exception: SM 326; DI 8 x NAMUR, for which this parameter does not exist. For this module, only 2-channel equivalent sensor interconnection can generally be selected where "1oo2 evaluation" is concerned.):

- "2-channel equivalent":

A two-channel sensor or two single-channel sensors (2-channel connection) is/are connected to two opposite input channels.

- "2-channel non-equivalent":

One non-equivalent sensor or two single-channel sensors (2-channel non-equivalent) is/are connected to two opposite input channels.

- "Single-channel":

One sensor (1-channel) is connected to two opposite inputs.

Note

If you are using the "2-channel non-equivalent" or "1-channel" type of sensor interconnection, and the sensor is being supplied by the F signal module, the left-hand side of the F signal module must use supply voltage Vss.

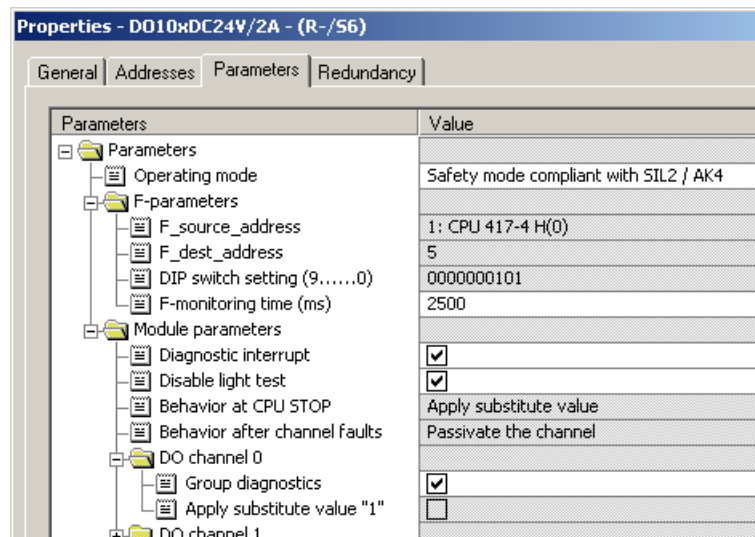
Discrepancy time

Where "1oo1 evaluation" is concerned, the value displayed is not relevant.

The discrepancy analysis for equivalence/non-equivalence is used for fail-safe inputs in order to detect errors from the temporal characteristic of two signals with identical functionality.

The discrepancy analysis is started whenever different levels (when testing for non-equivalence: the same level) are detected on two associated input signals. A test is run to see whether, once a configurable period of time known as the discrepancy time has elapsed, the difference (when testing for non-equivalence: the match) disappears. If not, there is a discrepancy error.

3.4.6 Assigning parameters for the SM326; DO 10 x DC 24V/2A



On fail-safe output modules, the required safety class is achieved by injecting test signals.

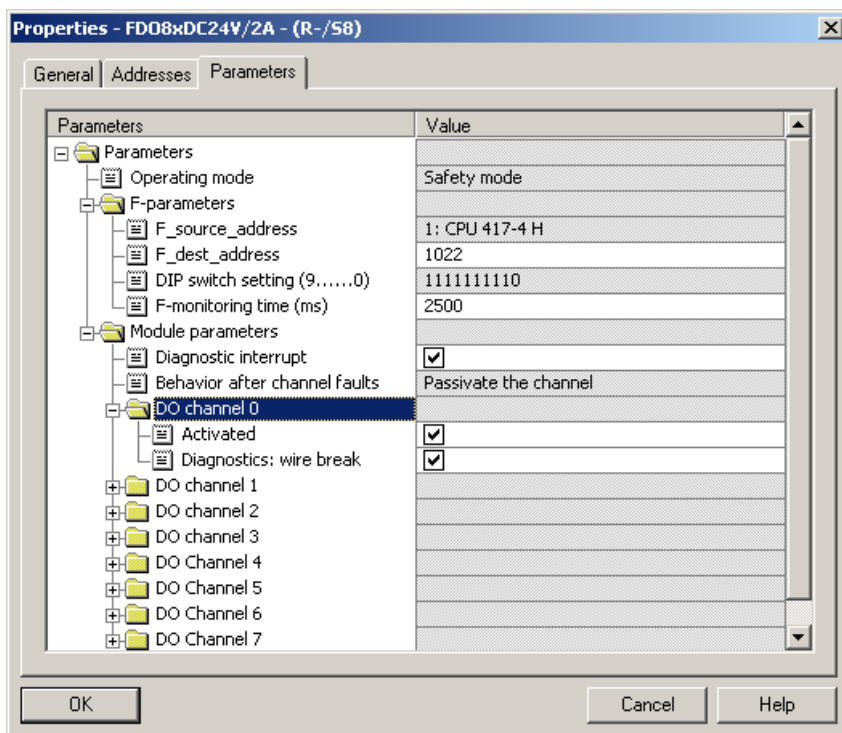
Light test

For the purpose of the test, 1-signals are connected to the output while the output is inactive (output signal "0"). This setting activates the output briefly (< 1 ms) (= "light period").

The dark test, which the module performs cyclically, is sufficient for SIL 2. This involves connecting 0 signals to the output while it is active. The output is deactivated briefly (< 1 ms) ("dark period") in order to detect short circuits. In order to detect cross circuits between outputs, during the dark test various bit patterns are issued one after the other to a group of outputs (first the left half of the module, then the right half).

For SIL 3, the light test also needs to be performed or the output switched at least once a day.

3.4.7 Assigning parameters for the SM326; DO 8 x DC24 V/2A PM



The module can only be used in safety mode, not redundantly. For the purpose of switching an actuator, each module is provided with one switch in the plus line (P switch) and one in the minus line (M switch). An actuator must be connected between the P and M switches to enable the module to be used for safety applications up to SIL 3.

Diagnostic interrupt

The diagnostic interrupt for the F signal module must always be activated in safety mode.

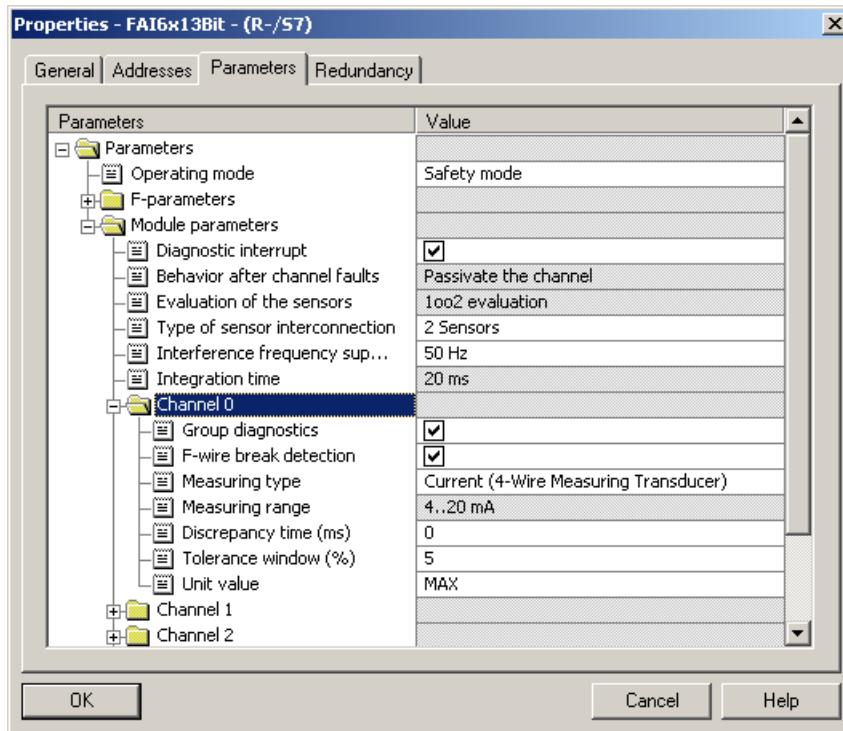
Activated

Activates channel processing

Diagnostics: Wire break

Activates wire-break monitoring on the channel

3.4.8 Assigning parameters for the SM336; AI 6 x 13Bit

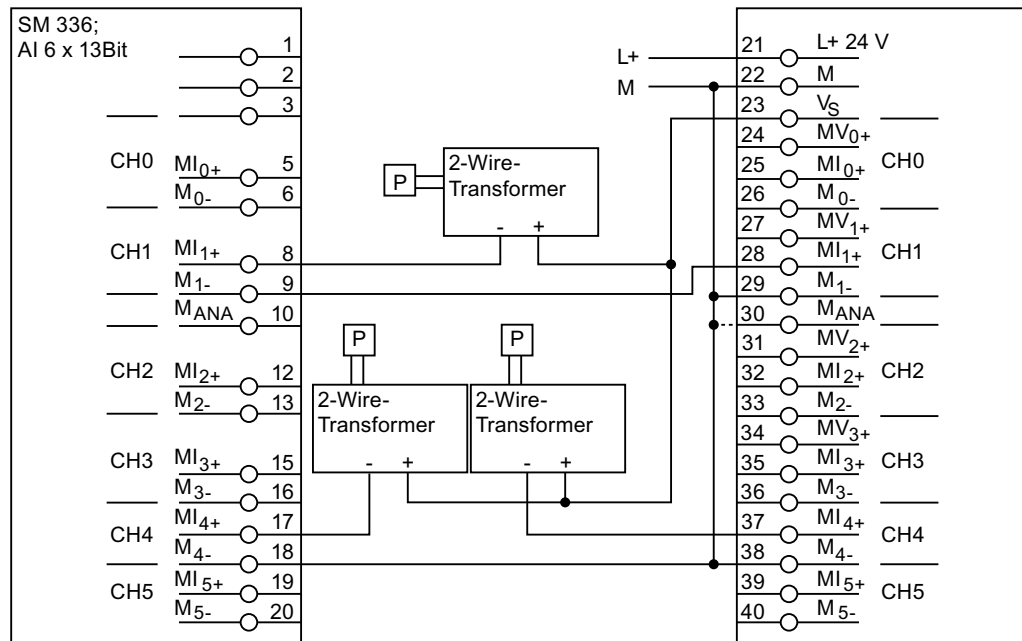


Sensor evaluation (analog inputs)

- 1oo2 evaluation

1 sensor connected to the module via a single-channel redundant connection (voting on module). The module has 6 redundant SIL 3-compatible channels.

Safety class SIL3 can be achieved here:



Type of sensor interconnection (analog inputs)

When safety mode is activated 1 or 2 sensors can be configured per input channel. Discrepancy handling can be set accordingly.

Interference frequency suppression

Setting interference frequency suppression for the line frequency The corresponding integration time of the analog digital converter is displayed.

If you change this setting, the increment for the F monitoring time and for the discrepancy times will also change automatically. The values set there will be adjusted to the next lowest value.

F wire-break detection

You can set whether a wire-break check is to be performed for each individual channel (< 3.6 mA; otherwise, detected at 1.18 mA underflow).

If wire break is detected, a diagnostics interrupt will be triggered in the CPU and a corresponding process control message will be sent on the OS.

F-short-circuit detection

If a short-circuit is detected, a diagnostics interrupt will be triggered in the CPU and a corresponding process control message will be sent on the OS.

Advanced short-circuit diagnostics can be triggered when required by means of additional limit-value monitoring.

Measurement type

You can select the measurement type for a channel or deactivate a channel (both on a channel-specific basis) here.

- "U" for voltage measurements (only possible for channels 0 to 3 and only in standard mode)
- "2DMU" or "4DMU" for current measurements (as appropriate for the transducer used).
- "Deactivated": The channel is not processed by the module.

The measurement type depends on the operating mode selected.

Safety mode	Voltage measurement	Current measurement
activated	Not possible	2DMU, 4DMU
Not activated	possible	2DMU, 4DMU

Measuring range

The selection options in the measuring range field vary depending on the selected mode (safety mode activated or deactivated) as well as the measurement type. If a channel is deactivated, it will not be possible to select a measuring range.

Note

If you are using Marshallled Termination Assemblies (MTAs), you should select 4DMU for the configuration, since the supply is provided via the MTAs.

The following measuring ranges can be selected in safety mode:

Measurement type		Range	Channels
2DMU	Current measurement 2-wire transducer	4 to 20 mA	0 to 5
4DMU	Current measurement 4-wire transducer	4 to 20 mA	0 to 5

Note

In safety mode, only the 4 to 20 mA measuring range is permitted. In this measuring range, a current of < 3.6 mA will produce a wire-break signal. If, however, the type of sensor you are using (e.g. a gas sensor) means that you do need to process signals that fall below this range, you can deactivate F wire-break detection. In this case, a current of < 1.18 mA will produce an underflow message.

For more information, please refer to the following FAQ: "How can process signals that are less than 4 mA be used with a 4 to 20 mA analog input module (F technology)?"
(<http://support.automation.siemens.com/WW/view/en/23707366>)

Discrepancy handling (analog inputs)

In the processing industry, no evaluation is generally performed between 2 signals on the module. 1oo1 is set for sensor evaluation. This makes all the signals available in the user program, where they can be linked in 1oo2 or 2oo3, depending on what is required. If 1oo2 evaluation is to be implemented on the module, you can find the parameter description for discrepancy handling in the online help.

Note

For details of possible types of interconnection, please refer to the FAQ titled "Wiring & Voting Architectures for ET 200M F-AIs".
(<http://support.automation.siemens.com/WW/view/en/24690377>)

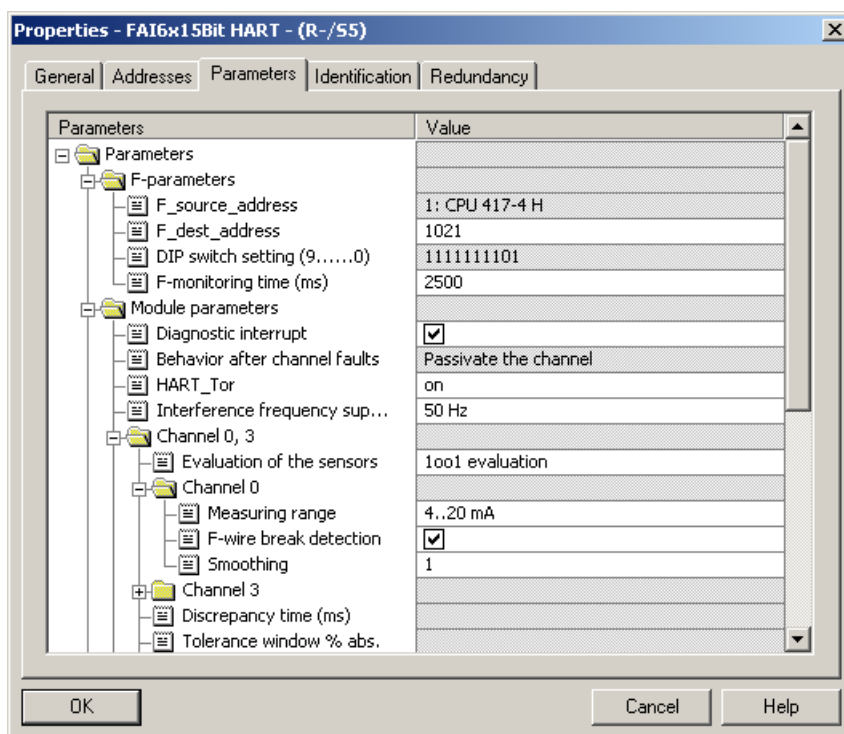
3.4.9 Assigning parameters for the SM336; F-AI 6 x 0/4...20mA HART

Note

In order to use this module in SIMATIC PCS 7, you require additional software that varies depending on the version of PCS 7 you are using. This is available for download from the Service & Support portal:

Also refer to:

Software components (Page 9)



Diagnostic interrupt

The diagnostic interrupt for the F signal module must always be activated in safety mode.

HART gate

This enables HART communication with the transducers to be controlled. ON/OFF switches HART communication on or off for the entire module, in a safety-related manner. If "Can be switched" is set here, HART communication may be enabled or disabled on the F channel driver.

Interference frequency suppression

Setting interference frequency suppression for the line frequency. The corresponding integration time of the analog digital converter is displayed.

If you change this setting, the increment for the F monitoring time and for the discrepancy times will also change automatically. The values set there will be adjusted to the next lowest value.

Sensor evaluation

- 1oo1 evaluation: Each channel is considered individually and the input value forwarded to the CPU.
- 1oo2 evaluation: 2 channels are combined in all cases (0/3, 1/4, and 2/5). A discrepancy analysis is performed on the module and the configured input value is forwarded to the CPU. With this setting, the parameters for the discrepancy analysis can be configured.

Measuring range

The measuring ranges 0 to 20 mA and 4 to 20 mA are available for selection. With 0 to 20 mA, HART communication is not possible.

F wire-break detection

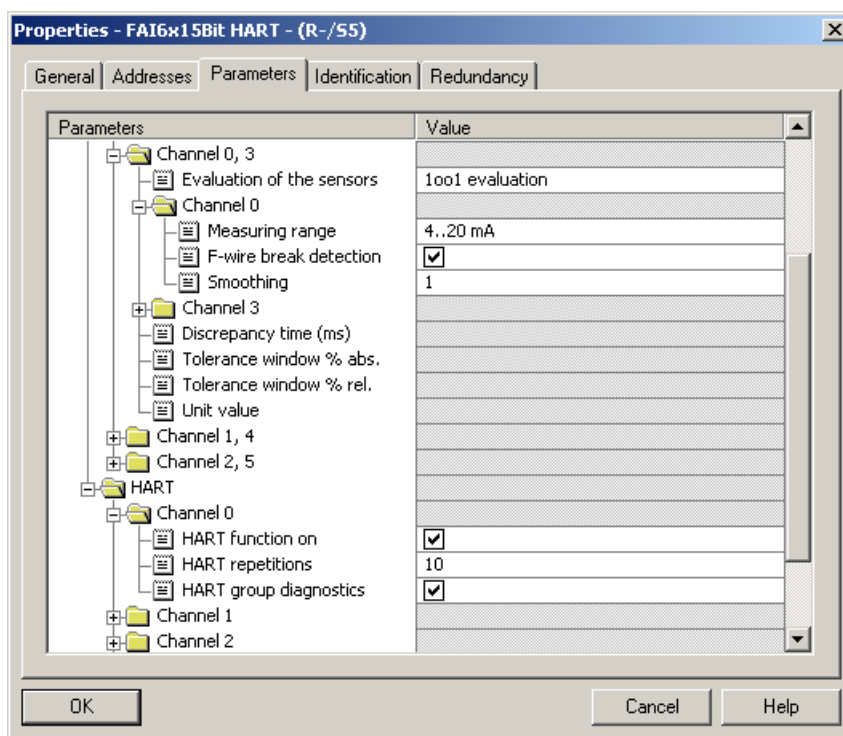
In the 4 to 20 mA measuring range and with wire-break detection activated, a message is issued when a current of < 3.6 mA is present. If wire-break detection is deactivated, an underflow message is issued if a current of < 0.4444 mA is present (as with the 0 to 20 mA measuring range).

Filter

The module filters the input signal throughout the specified number of acquisition cycles. Please note that input signal filtering will lengthen the system's reaction time.

Discrepancy handling (analog inputs)

In the processing industry, no evaluation is generally performed between 2 signals on the module. 1oo1 is set for sensor evaluation. This makes all the signals available in the user program, where they can be linked in 1oo2 or 2oo3, depending on what is required. If 1oo2 evaluation is to be implemented on the module, you can find the parameter description for discrepancy handling in the online help.



HART

Here, you can switch off HART communication on specific channels in a non-safety-related manner, as well as enable HART diagnostics options and determine how often the module will attempt to establish HART communication with the transducer before a message is issued.

In the case of HART devices for which parameter assignment cannot be interlocked, HART communication must be switched off.

3.5 Configuring redundant F-signal modules

You can use the fail-safe signal modules S7-300 (F-signal modules) redundantly in one or several different ET 200Ms. Where F signal modules configured with redundancy are concerned, please note the following:

- The two F signal modules are of the same product version.
- "Safety mode" is set for both F signal modules.

Where fail-safe signal input modules configured with redundancy are concerned, it is possible to draw a distinction between two application cases:

- One sensor: The sensor is wired to both redundant F signal modules.
- Two sensors: One sensor is wired to each of the redundant F signal modules (module and sensor are redundant).

For redundant fail-safe digital input modules, the F channel driver F_CH_DI can run a discrepancy analysis to increase availability when S7 F Systems Lib V1_3 is being used. You need to set the "Discrepancy time" parameter in the hardware configuration for this purpose. Set a discrepancy time of "0" to deactivate the discrepancy analysis.

The F channel driver F_CH_DI provides the result of the discrepancy analysis at output DISCF or DISCF_R. The signals must either undergo further processing in the logic or be reported with a message block, e.g. MESSAGE.

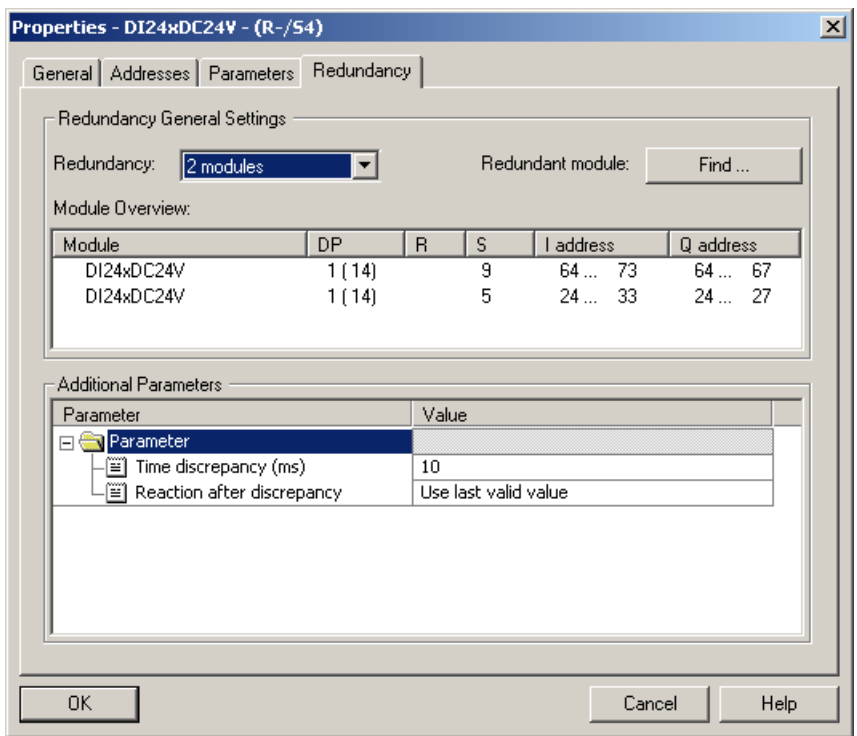
The F_CH_AI does not perform a discrepancy analysis. Wiring one sensor to both F signal modules involves the use of Zener diodes. If you require a discrepancy analysis for the purpose of monitoring Zener diodes, select the same procedure as for two sensors.

If you are using two sensors, then both signals should also be available in the user program. In this case, it is not possible to use the redundancy function of the F-AI module. Read both signals in and use block F_1oo2AI for signal selection and discrepancy analysis purposes.

Procedure

1. In HW Config, configure both F signal modules in the ET 200M stations.
2. Configure the first F signal module:
Activate "Safety mode" in the "Parameters" tab.
3. Configure the second F signal module:
Activate "Safety mode" in the "Parameters" tab.
4. For the second F signal module, select "2 modules" mode in the "Redundancy" tab.
5. In the "Find redundant module" dialog for the F signal module, select the redundant F signal module.
6. Set further parameters as necessary. The settings are applied automatically for the redundant F signal module. As soon as two F signal modules are redundant, changes to the parameter settings for one of them will automatically be applied for the other.

7. Check the default discrepancy time for redundant, fail-safe digital input modules.



8. Create a symbol for the lower I/O address and interconnect the channel driver with this address.

If you are operating a HART device on a channel of a redundant module, you also need to follow the steps outlined below:

1. In both modules, configure a "HART field device" on the relevant channel.
2. In the properties, select "On" or "Can be switched" under the HART_gate parameter and activate the HART function of the relevant channel with the field device.
3. Double-click the HART field device of the module with the lower I/O address and configure the field device in PDM.
4. When you save your settings in PDM, they will all be applied to both modules.

Note

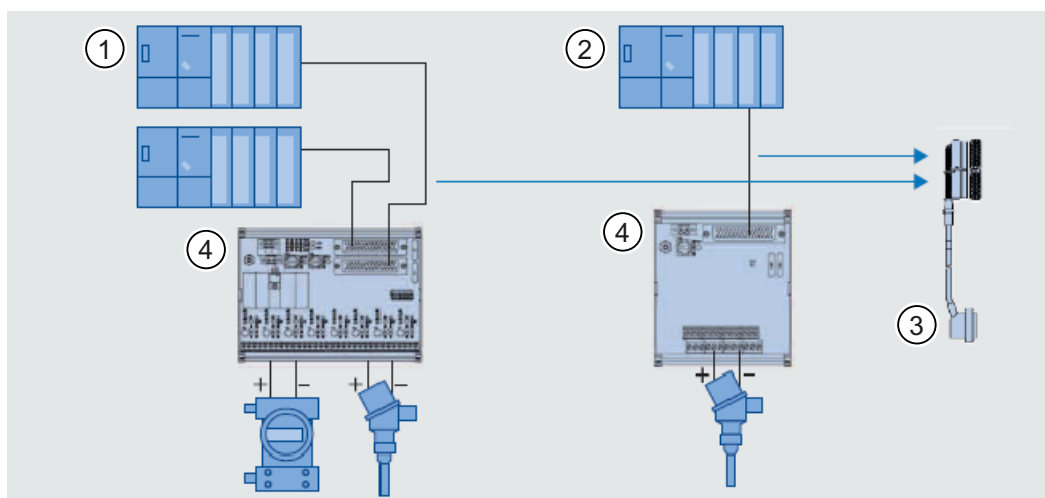
For more information on configuring redundant I/O modules, please refer to "PCS 7 Compendium Part A - Configuration Guidelines".

3.6 Marshalled Termination Assemblies (MTAs)

The MTA Terminal Modules enable you to connect field devices, sensors, and actuators quickly and easily to the F signal modules of the ET 200M distributed I/O. MTAs are available for standard and F signal modules.

Pre-assembled cables are used to wire the MTAs to the singular or redundant ET 200M modules. MTAs can be used to significantly reduce the time and money spent on cabling and commissioning; they also help to avoid wiring errors.

The figure below shows how an MTA is incorporated into the automation system. Integration can be singular and redundant.



①	ET 200M, redundant
②	ET 200M, singular
③	Prefabricated cable with front connector
④	MTA

Note

For more information, please refer to the manual titled *"ET 200M Marshalled Termination Assemblies Remote I/O Modules"*.

(<http://support.automation.siemens.com/WW/view/en/22091986>)

3.7 "Wiring and Voting" architectures for ET 200M

Note

For details of possible types of interconnection, please refer to the FAQ titled "Wiring & Voting Architectures for ET 200M F-AIs".

(<http://support.automation.siemens.com/WW/view/en/24690377>)

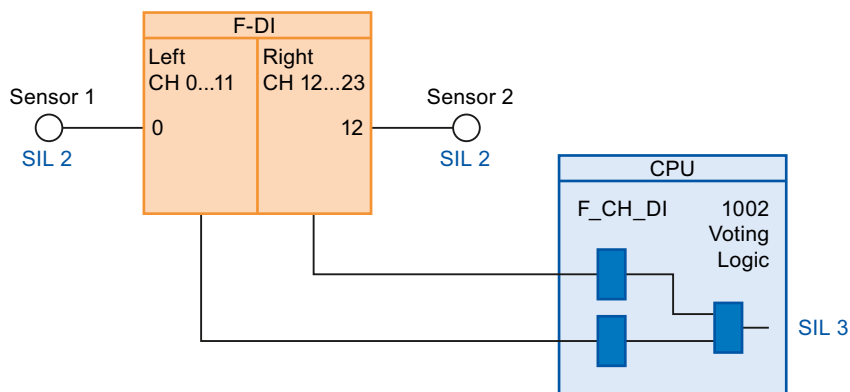
1oo2 voting of fail-safe input signals can be implemented in both the F signal module and the user program.

2oo3 voting of three fail-safe signals can be implemented in CFC using a function block, or in the Safety Matrix.

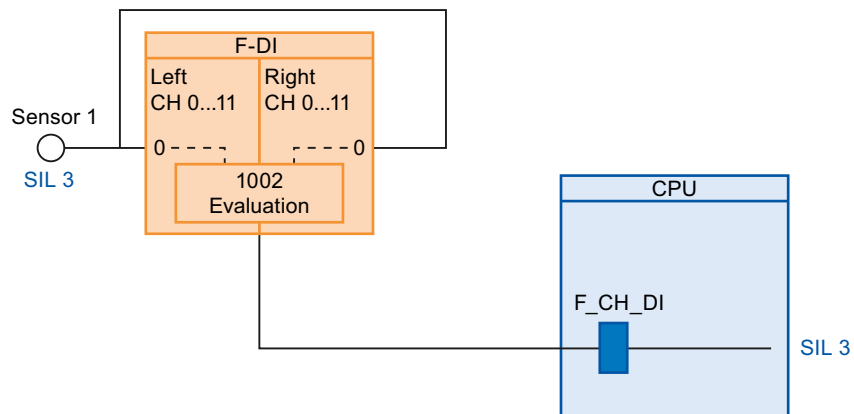
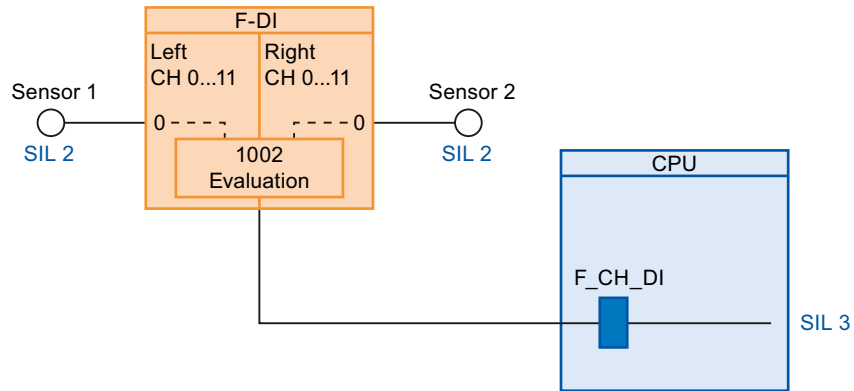
3.7.1 Voting with F-DI

Various types of wiring are shown in the examples below.

SIL 3 by means of voting in the F user program



SIL 3 by means of voting in module

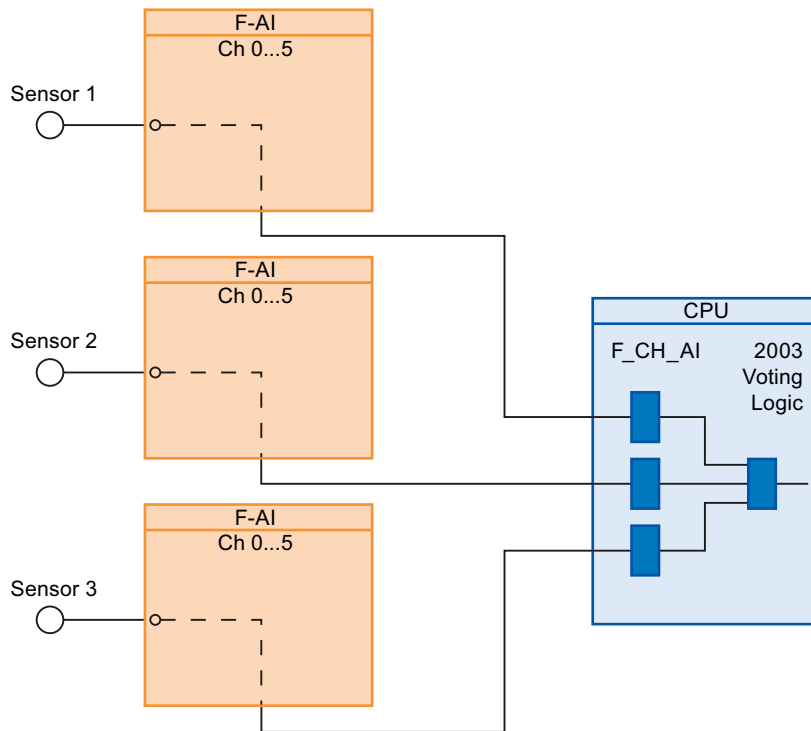


NOTICE
Suitable sensors are required to achieve this SIL3 wiring.

3.7.2 Voting with F-AI

2-out-of-3 selection uses three sensors and, for example, three F-AI modules.

In the example, each sensor is wired to channel 0 of an F-AI module. The individual signals are then evaluated in the user program. SIL 3 can be achieved with a 2-channel configuration.

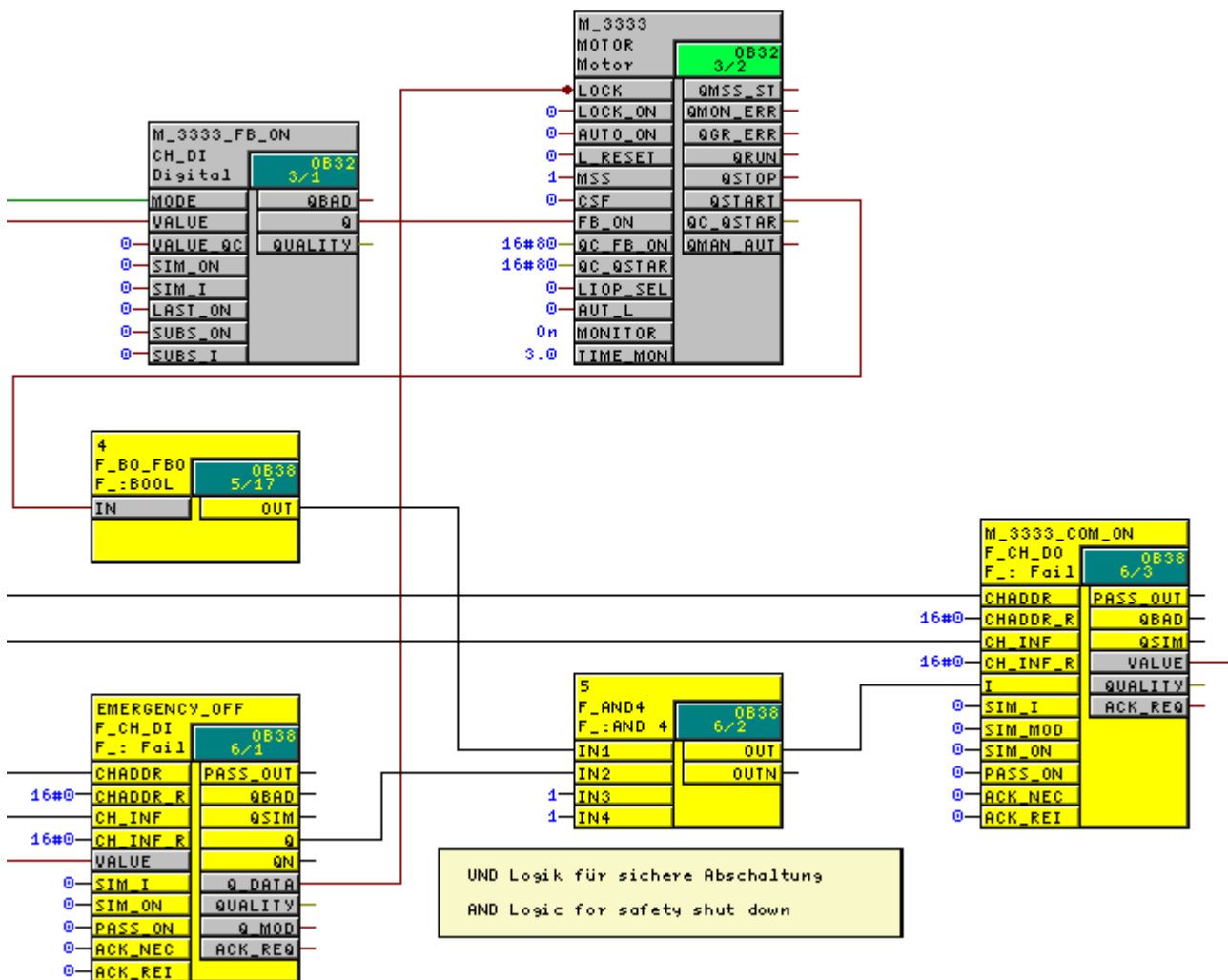


Configuring the safety program

4.1 Introduction

Fail-safe user program

Use the F blocks supplied in a library with the S7 F Systems optional package to create a fail-safe user program (F program or safety program) with CFC.



As well as functions for programming safety functions, the F blocks contain functions for detecting and reacting to errors. In other words, they ensure that failures and errors are detected and that an appropriate reaction is triggered to maintain the F system in or switch it to a safe state. The safe state in a fail-safe function is generally "0"; i.e. a "0" signal at an input or output leads to disconnection.

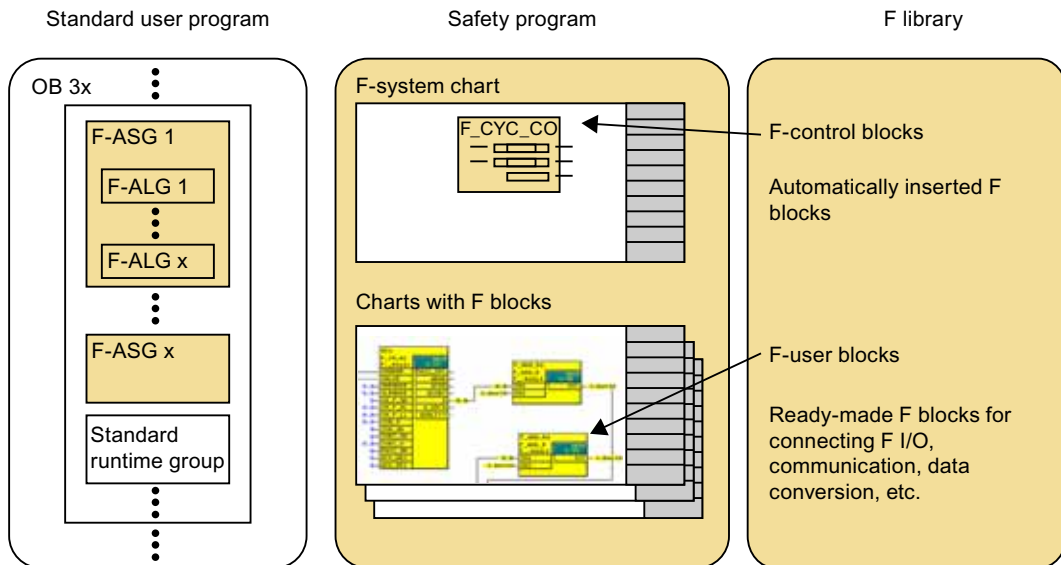
The user program in the CPU can be created from F and non-F blocks. The F program is configured in separate CFC charts.

Data transfer between the standard program and the F program is handled using conversion blocks. Please note that safety functions must be implemented with F blocks consistently from the input driver to the output driver. Standard signals may only change the state of a safe output if the safety function is in a healthy state.

During compilation, specific functions for detecting and reacting to errors are automatically added to the F program. The S7 F System optional package also features functions for comparing F programs and providing support for the acceptance and approval procedure for F programs, such as functions for generating a signature via the F program which can be used to detect changes to functions and parameters. This signature is saved during the plant acceptance procedure.

Program structure of the safety program

The figure below shows a diagram illustrating the structure of a safety program comprising CFC charts with F blocks which are assigned to F-runtime groups.



Some of the properties possessed by the various components are:

- The safety program contains F runtime groups and the charts assigned to them. The charts contain F blocks with their parameter settings and interconnection.
- The F-runtime groups are added to one or more cyclic interrupt OBs.
- F-runtime groups can be combined in F-shutdown groups (F-SG).
- The cyclic interrupt OB can also contain standard runtime groups.
- The F blocks in the S7 F Systems F library appear in yellow on the CFC chart in order to highlight the fact that there is a safety program involved.
- The CFC charts and F runtime groups with F blocks appear in yellow and are marked "F" in order to distinguish them from charts and runtime groups associated with the standard user program.

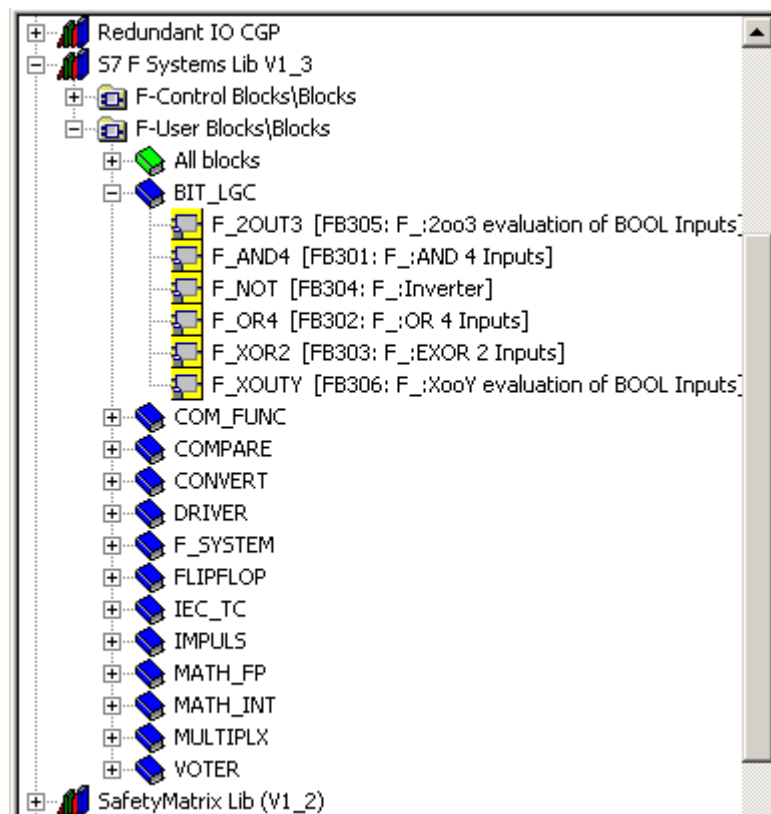
4.2 Creating the safety program

Requirements

- You must have created a project structure in the SIMATIC Manager.
- Prior to programming, you must have configured the hardware components of your project, in particular the F CPU and the F signal modules, for safety mode.
- You must have assigned your safety program to a 412-3H CPU, 414-4H CPU, or 417-4H CPU.

Library

The F-user blocks must always be used for configuration.



4.2.1 Defining the program structure

In addition to considering the standard scenario, you need to answer the following questions when drafting a safety program:

- Which parts of the user program need to be fail-safe?
- What are the response times you wish to achieve?

You will need to split your F program into various cyclic interrupt OBs (OB 30 to OB 38) in accordance with these requirements.

You will improve performance if you program parts of the program which are not needed for the safety functions in the standard user program.

In terms of dividing your program between the standard user program and the safety program, please remember that the standard user program is easier to change and download to the F CPU. Changes to the standard user program do not usually need to undergo acceptance and approval.

Rules governing program structure

When drafting a safety program for S7 F/FH Systems, you need to observe the following rules:

- F-runtime groups with F blocks can only be assigned to cyclic interrupts OB 30 to OB 38.
- A chart can contain both F blocks and standard blocks; they need to be inserted in separate runtime groups in this case. You are not permitted to use these charts as F block types.
- In the safety program, access to the F signal modules is only permitted via the F channel driver (F_CH_xx).

4.2.2 Creating CFC charts

Inserting CFC charts

Individual CFC charts are added to the chart folder or plant hierarchy (PH) in the same way as for standard user programs:

- In the chart container "Insert New Object > CFC" in the SIMATIC Manager
- Directly in the PH or process object view in the relevant hierarchy folder "Insert New Object > CFC"

Inserting F blocks

Blocks are dragged from the "S7 F Systems Lib V1_3" library ("Failsafe Blocks (V1_2)" > "F-User Blocks") and dropped into the chart. There is no limit on the number of times a block can be dragged and dropped.

Note

If a block type has been dragged and dropped from the library before, the process can be completed more quickly the next time by using the "CFC block catalog", "Blocks" tab.

4.2.3 Assigning parameters to and interconnecting F-blocks

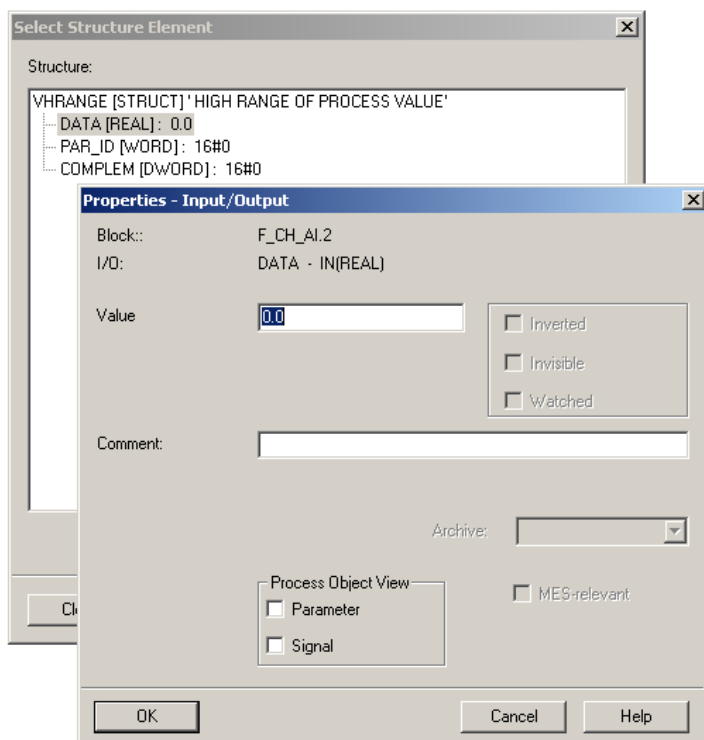
F block inputs and outputs are parameterized and interconnected using the standard CFC procedure.

NOTICE

You are not permitted to interconnect EN/ENO connections of F blocks and F runtime groups. You may not assign a value of 0 (FALSE) to EN I/Os either.

Special F data types in a safety data format are used for fail-safe block connections. The safety data format enables data and address errors to be detected.

In terms of programming, the F data types are implemented as structures in which only the "DATA" component is ever relevant for the user.



Example: Structural element F_Real VHRANGE [STRUCT]"HIGH RANGE OF PROCESS VALUE"

You can change the structure comment to whatever you wish.

If you wish to change the value (default) of a block connection with an F data type, you may only change the DATA component.

Changes to the input parameters of F blocks with F data types can be made as follows:

- Offline with the assistance of the CFC editor
- Online using CFC test mode with safety mode deactivated

Note

Values of PAR_ID and COMPLEM must not be changed.

If errors in the safety data format are detected during the execution of the safety program, an F-STOP is triggered.

4.2.4 Run sequence of F-blocks

Defining the run sequence

You define the run sequence in the CFC editor in the same way as for a standard user program. Changing the run sequence also changes the collective signature.

Correct run sequence of F blocks

The sequence of the F blocks within the F shutdown group is relevant. The number of F-runtime groups the F-shutdown group has been split into is of no relevance.

Essentially, the correct run sequence of the various F block types is as follows:

1. Placed automatically:
 - F module driver for F signal modules with inputs or with inputs and outputs
 - F-communications blocks and F system blocks for receiving
 - F blocks for data conversion from standard value to F structure
2. F channel drivers for inputs
3. F blocks for user logic

4. F channel drivers for outputs
5. Placed automatically:
 - F block F_PLK
 - F block F_PSG_M
 - F module driver for F signal modules with outputs or with inputs and outputs
 - F-communications blocks and F system blocks for sending
 - F block F_PLK_O
 - F block F_DIAG (S7 F Systems Lib V1_3 and higher)

The run sequence of the blocks listed under items 1 and 5 is adjusted automatically when the F program is compiled.

The IPO principle (input, process, output) must always be observed when placing F channel drivers and F blocks for user logic. This ensures that all inputs are read first, the relevant processing steps are performed, and all outputs are then written.

Furthermore, F monitoring blocks only visible following compilation are also added automatically.

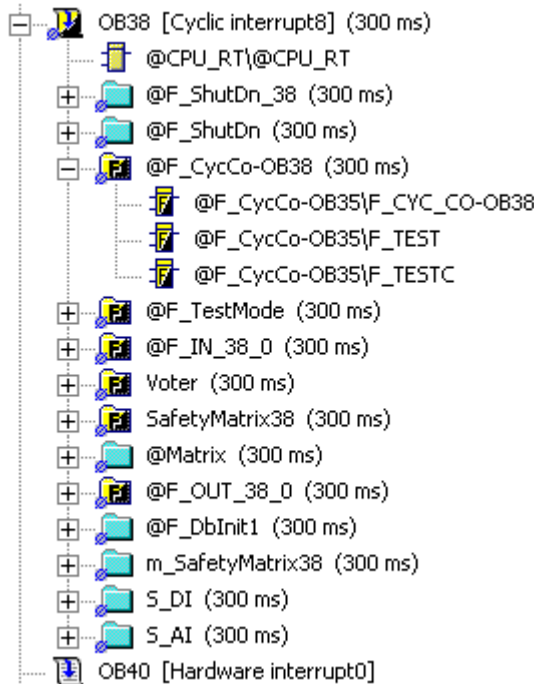
Note

You are not permitted to change the run sequence of automatically inserted F-control blocks. Neither can these blocks be deleted or changes made to them.

4.2.5 F-runtime groups

During the programming of the safety program, F blocks cannot be inserted directly into tasks (cyclic interrupt OBs). When a new CFC chart is created in PCS 7, the system will automatically generate a runtime group of the same name, into which the F blocks placed in the corresponding CFC chart can then be inserted.

An F-runtime group only becomes an F-runtime group (identified by a yellow folder and "F") when F blocks are called in it.



With F Systems Lib V1_3 and higher, separate runtime groups are created for the input and output drivers the first time compilation takes place.

Runtime groups of an F program

The F program is divided into several runtime groups, as shown in the table below.

Note

xx = Number of the cyclic interrupt OB

y = Consecutive numbering if several shutdown groups exist in a single cyclic interrupt OB

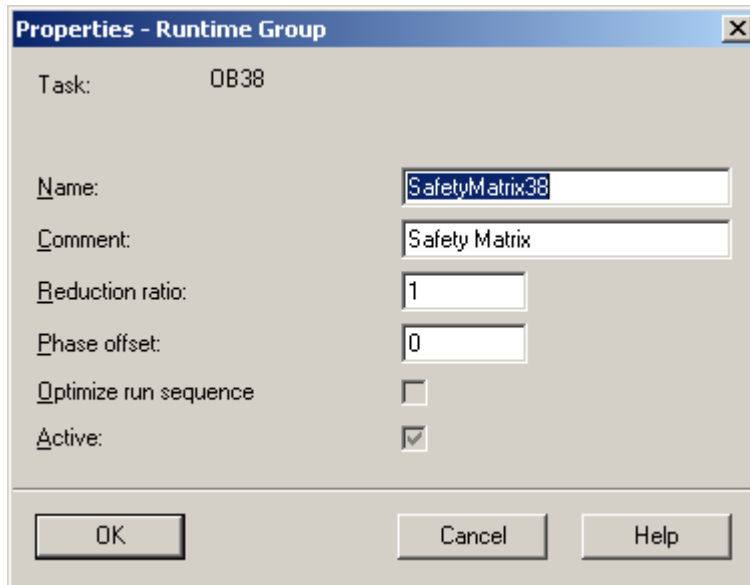
Chart	F blocks
@F_ShutDn_xx	Shutdown logic of cyclic interrupt OB The shutdown logic is created with RTGLOGIC and standard logic blocks.
@F_ShutDn	Shutdown block Where an F program is present in several cyclic interrupt OBs, this runtime group is integrated into the cyclic interrupt OB with the shortest call time. The shutdown logic is created with F_SHUTDOWN, RTGLOGIC, and standard logic blocks.
@F_CycCo-OBxx	F_CYC_CO, F_TEST and F_TESTC (for tests)
@F_TestMode	F_TESTM for managing safety mode
@F_IN_xx_y	F blocks which supply input values for the F program (F_QUITES, communication receive blocks, conversion blocks from standard to F data types, drivers for F input modules)
	Runtime groups with user logic, in the sequence in which they were created by the user
@F_OUT_xx_y	Blocks which further process output values of F blocks (communication send blocks, conversion blocks from F to standard data types, drivers for F output modules, F_PLK, F_PLK_O, and F_DIAG for program sequence control)
@F_Dblnitxx	DB_INIT function block required for the cold restart of an F runtime group

All required error OBs are added to the block container in the SIMATIC Manager.

Rules for F runtime groups in the safety program

- We recommend that you proceed as follows in order to make the lengths of the F cycles as uniform as possible:
If you mix F and standard runtime groups in a cyclic interrupt OB, you must execute the F-runtime groups before the standard runtime groups.
- An F-runtime group must retain the default settings for the runtime properties, as follows:
 - Reduction ratio = 1
 - Phase offset = 0

You are not permitted to modify these values.



- You are not permitted to move automatically generated F runtime groups (identified by @). You are not permitted to move any @blocks within these automatically generated F-runtime groups either.

Note

The feature Optimize Run Sequence in CFC can lead to a change in the collective signature and impair the response times of the safety program, and therefore should not be used in F runtime groups.

With PCS 7 V7.0 SP1 and higher, it is no longer possible to optimize the run sequence for cyclic interrupt OBs with F-runtime groups.

4.2.6 F-shutdown groups

An F-shutdown group is a self-contained unit in your safety program. It contains user logic which is executed or shut down simultaneously. The F-shutdown group contains one or a number of F-runtime groups which are assigned to a common cyclic interrupt OB. You can choose whether an error during execution of the safety program will lead to a full shutdown or a partial shutdown of the safety program. With a partial shutdown, only the F-shutdown group in which the error occurred is shut down.

Rules for F shutdown groups in the safety program

You are not permitted to directly interconnect F blocks belonging to different F shutdown groups. Data can only be exchanged between F shutdown groups using special communication blocks (F_S_xx, F_R_xx). Therefore, all F channel drivers in an F signal module must be in the same F-shutdown group, together with the module driver of the F signal module.

Defining F shutdown groups

As soon as you place F blocks in the CFC editor for the first time, all of the F-runtime groups in a single cyclic interrupt OB will form an F-shutdown group.

You can configure each F-runtime group as the last F-runtime group in an F-shutdown group by placing the "selection block" F_PSG_M accordingly. The F system will then create a new F-shutdown group for all subsequent F-runtime groups until another F_PSG_M block is found.

Distribution/Combination by means of manual placing of F_PSG_M

If you add or delete one or a number of F_PSG_M blocks in your project, the order of your F shutdown groups will change. If you make a change to the layout of your F-shutdown groups, you must make sure that the F module drivers and all assigned F channel drivers are integrated in the same F-shutdown group.

You can split one F shutdown group into two F shutdown groups.

To do this, in the CFC editor's runtime editor, place the F_PSG_M block in the last F-runtime group which is to be assigned to the first F-shutdown group. All subsequent F-runtime groups will then be assigned to the second F-shutdown group.

The number of F-shutdown groups is limited to 110 in all cyclic interrupt OBs. The system restricts the number of F-runtime groups in an F-shutdown group.

You have the option of combining two F-shutdown groups. To do this, in the runtime editor of the CFC editor, delete the F_PSG_M block between the F-shutdown groups. If you combine a number of F-shutdown groups which exchange data via F communication blocks in a single F-shutdown group, you will need to remove these F communications blocks and replace them with direct interconnections.

Programming data exchange between F-shutdown groups

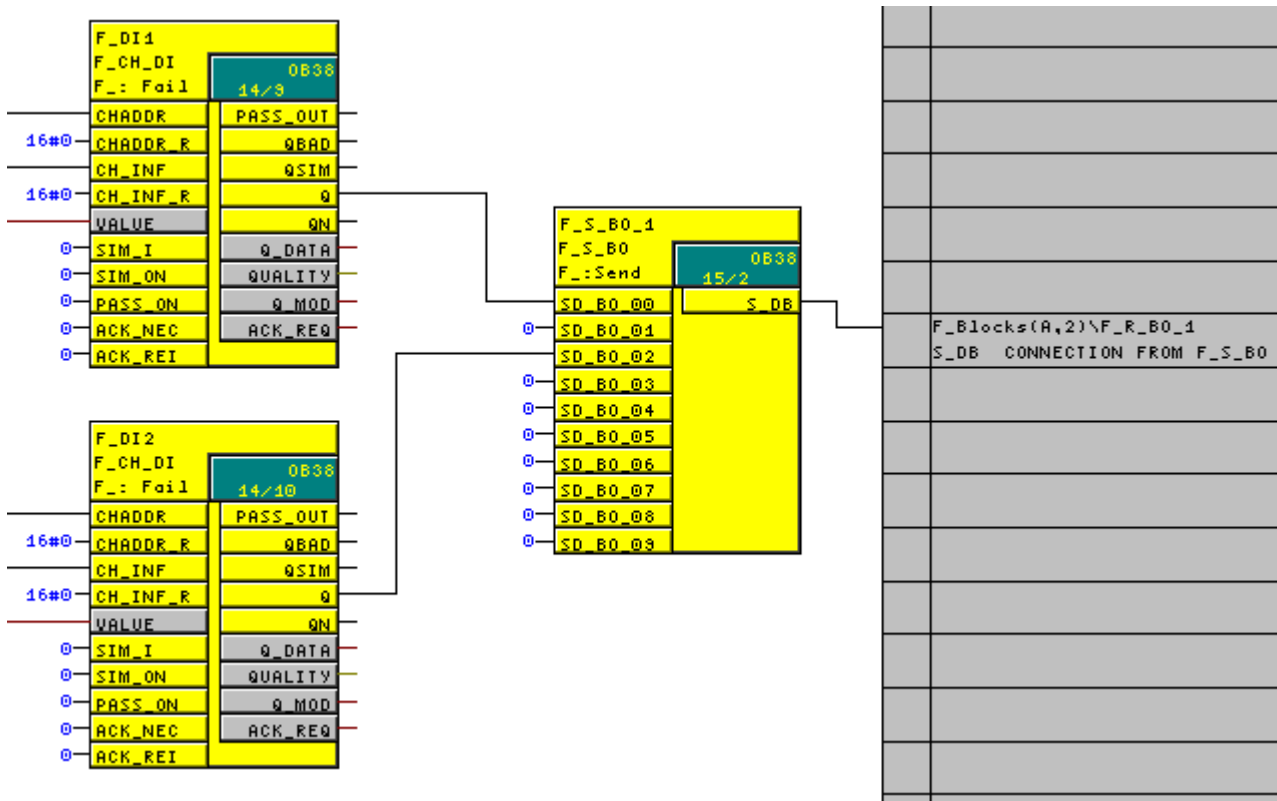
If you wish to exchange data between two F-shutdown groups, you are not permitted to interconnect the inputs and outputs directly. You need to use the following F-system blocks for data exchange between F blocks in different F-shutdown groups:

F block	Description
F_S_R/F_R_R	Safe transfer of 5 data of the F_REAL type
F_S_BO/F_R_BO	Safe transfer of 5 data of the F_BOOL type

Procedure for data exchange

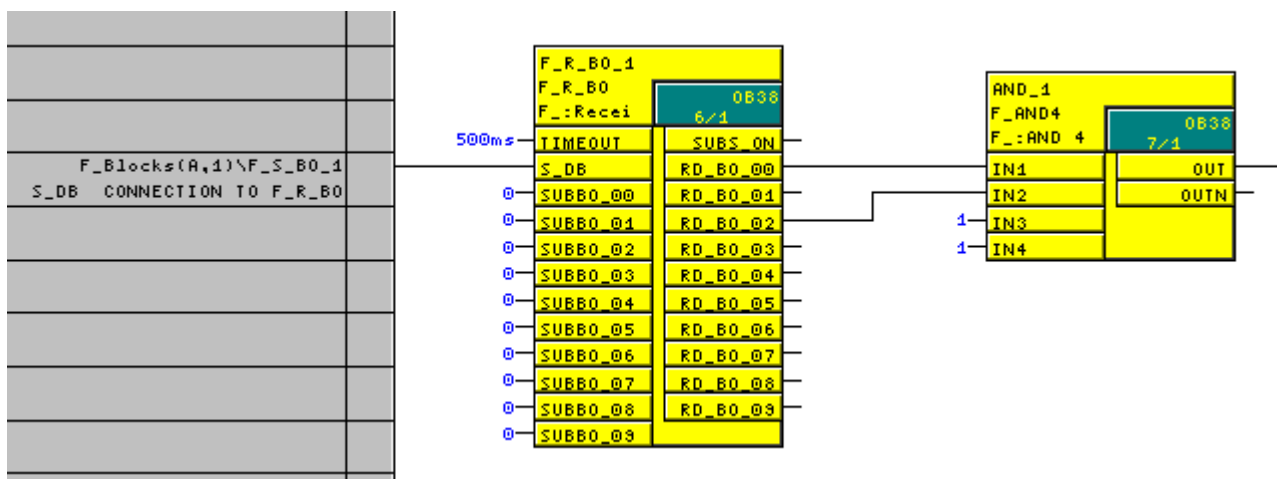
1. In the F-shutdown group from which data is to be transferred, add an F_S_R or F_S_BO type F block.
2. In the F-shutdown group to which data is to be transferred, add an F_R_R or F_R_BO type F block.
3. Interconnect the SD_R_xx inputs of the F_S_R or the SD_BO_xx inputs of the F_S_BO with the data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

Extract from a chart for shutdown group 1 with send block:



The connection to shutdown group 2 is established by linking output S_DB of block F_S_BO_1 in shutdown group 1 with input S_DB on block F_R_BO_1 in shutdown group 2.

Extract from a chart for shutdown group 2 with receive block:

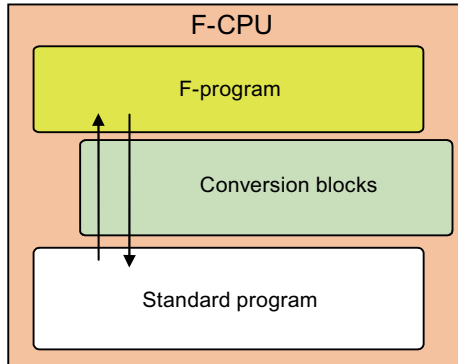


See also

Monitoring times and system response times (Page 110)

4.2.7 Data exchange between the F user program and standard user program

The standard program and the F program use different data formats. Accordingly, special conversion blocks have to be used for data exchange.

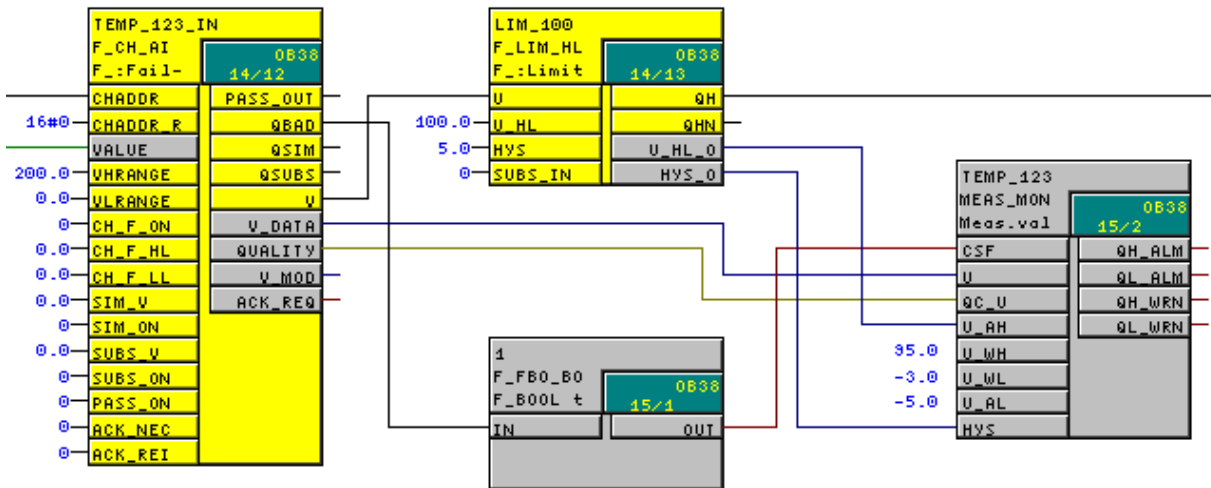


If you need the standard user program to process data from the F program further (for monitoring on the PCS 7 OS, for example), a block for F_Fdatatype_data type data conversion will have to be interconnected in the user program so that the F data types can be converted into standard data types.

Converting F data types to standard data types

The F blocks for converting F data types into standard data types (F_Fdatatype_datatype) must be called in the standard user program (standard runtime group).

Extract from a process tag chart; converting the "QBAD" signal from F_BOOL to BOOL:



Standard and F blocks are in different runtime groups.

As shown here, some F blocks have outputs (depicted in gray), which can be directly interconnected with standard PCS 7 blocks.

Standard PCS 7 blocks (e.g. MEAS_MON, DIG_MON, MESSAGE and their associated faceplates and process symbols) are used to visualize fail-safe analog values and status messages as well as system states and operating states.

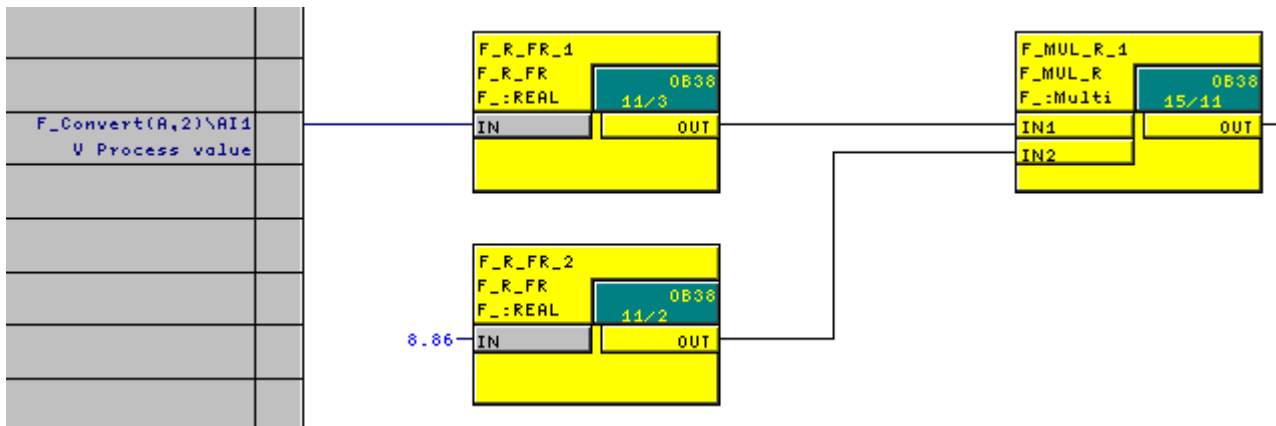
If parameters cannot be directly further interconnected due to the safety data format, the conversion blocks described above can be used.

Converting standard data types into F data types

If data from the standard user program is to be processed further in the F program, it will need to be converted.

The blocks for data conversion from standard data types to F data types (F_datatype_Fdatatype) can only be used in the F program (F-runtime group).

Extract from an F chart, conversion from REAL to F_REAL:



Note

The conversion blocks only perform data conversion; in other words, you will need to program additional measures in the F program for plausibility checking purposes (with F_LIM_R, for example), in order to ensure that only non-hazardous values are possible.

4.2.8 How F-blocks with floating-point operations respond to number range overflows

Within the context of analog value processing, number range overflows/underflows can occur during arithmetic calculations (with division by 0, root from a negative number or number range overflow).

With S7 F Systems Lib V1_3 and higher, the response is as follows in this case:

The results "Overflow (\pm infinite)", "Denormalized floating-point number" or "Invalid floating-point number (NaN)" are:

- Either output at the output and can be processed further by subsequent F blocks
- Or signaled to special outputs. A substitute value is output if necessary.

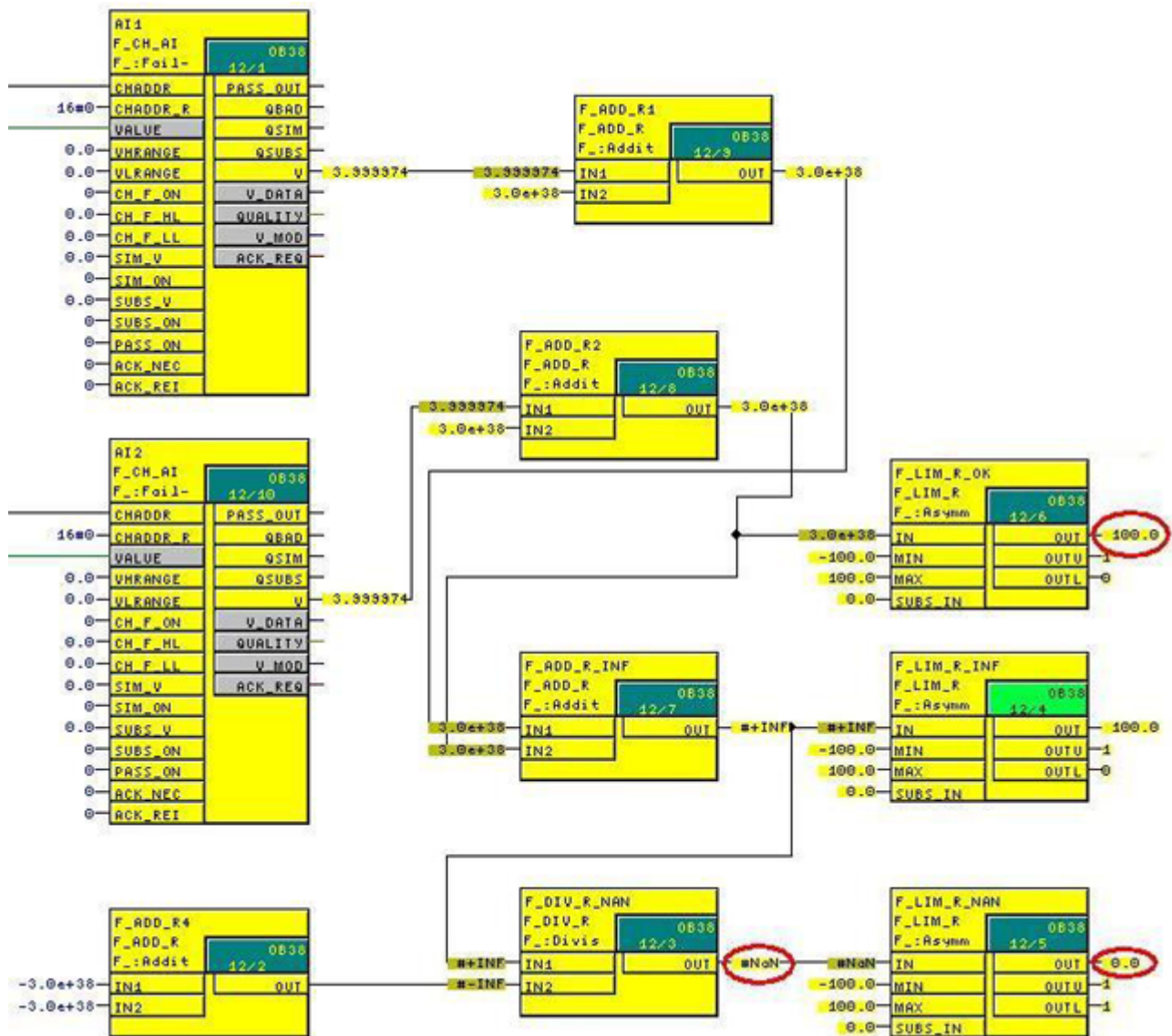
If the floating-point operation produces an invalid floating-point number (NaN) and no invalid floating-point number (NaN) existed as an address prior to this, the following diagnostics event will be entered in the F CPU's diagnostic buffer:

"Safety program: Invalid REAL number in DB" (event ID 16#75D9)

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

Configuring the safety program
 4.2 Creating the safety program

If you are not able to prevent these events from occurring in your safety program, you will need to decide, on the basis of your application, whether you wish to respond to them in your safety program.



You can use the F_LIM_R F block to check the result of a floating-point operation for overflow (\pm infinite) and invalid floating-point number (NaN).

- A limit violation is indicated by $IN > MAX$ or "+ infinite". MAX is output at OUT. OUTU is set to 1 and OUTL to 0.
- A limit violation is indicated by $IN < MIN$ or "- infinite". MIN is output at OUT. OUTU is set to 0 and OUTL to 1.
- If IN is between MIN and MAX, the input IN is forwarded to the output OUT. OUTU and OUTL are set to 0.
- If IN is an invalid floating-point number (NaN), the substitute value SUBS_IN is output at OUT. OUTU and OUTL are set to 1.

Note

You can find a detailed description of the F blocks in the help for the blocks, as well as in the *"S7 F/FH Systems Configuring and Programming"* Manual.
(<http://support.automation.siemens.com/WW/view/en/2201072>)

4.3 Configuring fail-safe AS-AS communication

Like standard communication, safety-related communication between the safety programs of F CPUs via S7 connections is implemented using connection tables in NetPro.

In S7 F/FH systems, safety-related communication via S7 connections is possible to and from the following F CPUs:

- CPU 412-3H
- CPU 414-4H
- CPU 417-4H

In the case of F Systems V6.0 with S7 F Systems Lib V1_3 and higher, fail-safe communication to S7 Distributed Safety is also supported with the following F CPUs:

- CPU 416F
- CPU 31xF

NOTICE

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

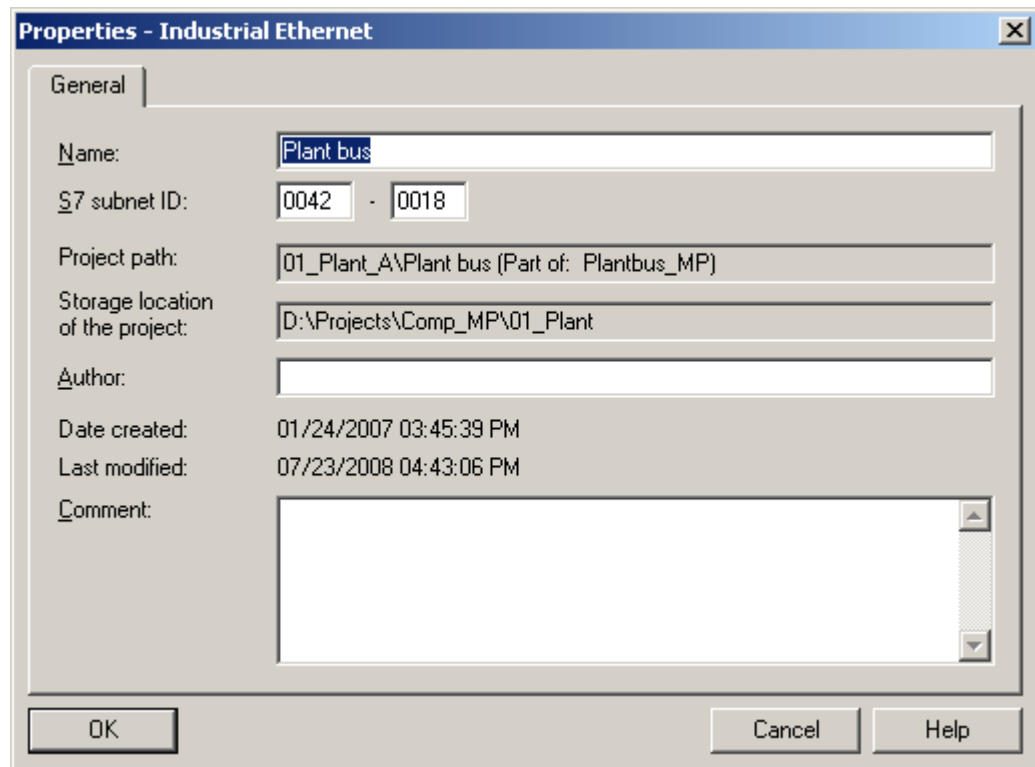
4.3.1 Configuring S7 connections

Configure S7 connections for safety-related AS-AS communication in exactly the same way as for standard communication (you might need also need to set up a fault-tolerant S7 connection).

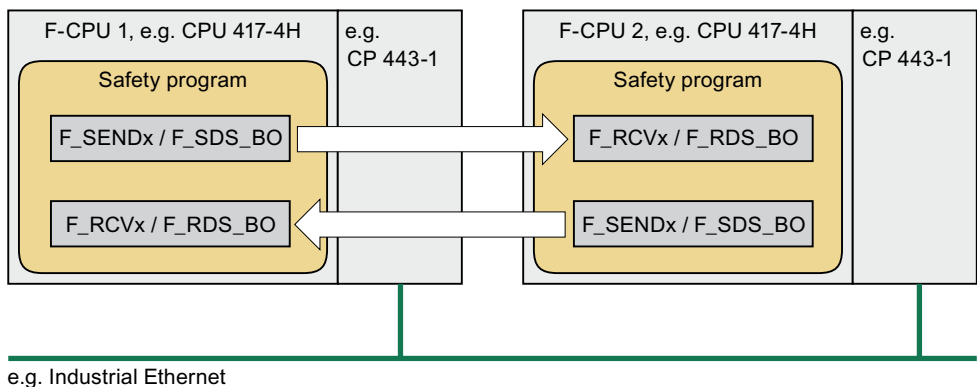
Note

You can find instructions on how to do this in the *"PCS 7 Compendium Part A - Configuration Guidelines"*.

During communication between automation systems in various subprojects, please make sure that the S7 subnet ID is the same in each of the respective projects. This ID is involved in the calculation of the CRC sum at the send and receive blocks and must, therefore, be the same in each case.



4.3.2 Configuring F-communications blocks



The following fail-safe blocks are available for communication between safety programs on various CPUs:

Block	Description
F_SENDBO/F_RCVBO	Safe transmission of 20 F_BOOL data type parameters
F_SENDR/F_RCVR:	Safe transmission of 20 F_REAL data type parameters
F_SDS_BO (F Systems V6.0 and higher)	Fail-safe sending of 32 F_BOOL data type objects to another F CPU
F_RDS_BO (F Systems V6.0 and higher)	Fail-safe reception of 32 F_BOOL data type objects from another F CPU

Requirements

The follow requirements must be fulfilled prior to configuration:

- The S7 connections between the F CPUs involved must be configured in NetPro.
- Both CPUs must be configured as F CPUs:
 - The password for the F CPU must be entered.
 - The "CPU contains safety program" option must be activated.

Procedure

Follow the steps outlined below:

1. Add the send block (F_SENDBO/F_SENDR) to the safety program from which data is to be transmitted.
2. Add the receive block (F_RCVBO/F_RCVR) to the safety program to which data is to be transmitted.
3. Assign the relevant IDs of the configured S7 connections to the "ID" inputs.
4. Parameterize the R_ID inputs. This defines the relationship between a send block and a receive block:

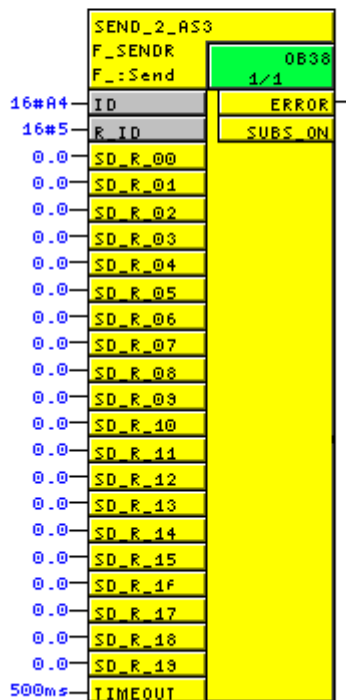
The associated fail-safe blocks are assigned the same (freely selectable, uneven value for R_ID. Please note that the R_ID+1 value is also assigned automatically.

Example: Local AS4 is sender, partner AS3 is recipient

Connection in NetPro AS4

Local ID	Partner ID	Partner
A4	A3	AS3_F / CPU 417-4 H

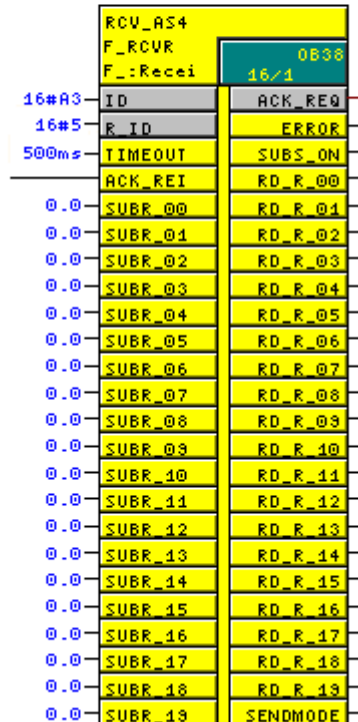
CFC Chart AS4



Connection in NetPro AS3

Local ID	Partner ID	Partner
A3	A4	AS4_FH / CPU 417-4 H

CFC Chart AS3



Note

If the R_ID is not an uneven number, the following error message will appear when the CFC charts are compiled:

"Module/connection with address/R_ID 16#0002/16#00000004 is being used by more than one block. [Assign a module/connection with this address/R_ID to no more than one block and use only uneven R_IDs.]"

5. Interconnect the ACK_REQ outputs of the F_RCVBO, F_RCVR, or F_RDS_BO F blocks to ascertain whether acknowledgment is required on reintegration following communication error elimination.
 6. Interconnect the relevant ACK_REI inputs of the F_RCVBO, F_RCVR, or F_RDS_BO F blocks with the signal for reintegration.
-

Note

If the S7 connections between the automation systems have been changed, the safety program will need to be recompiled.

User acknowledgment is always required for reintegration following PROFIsafe communication errors (ACK_REQ output set).

The communication load can be reduced if communication blocks are not processed unnecessarily in fast cyclic interrupt OBs.

For example, communication blocks could be processed in OB 37 (1 s) while faster F program parts run in a faster OB 38 (300 ms). If the process permits it, the timeout time should also be set as high as possible (e.g. 10 s).

If the maximum communication load of a system is, for example, 100 orders per second, this would represent a total of 20 communication orders with 5 F_RECVX and 5 F_SENDX blocks. If these are configured in OB 37 (1 s), this setting corresponds to a communication load of 20% based on the above assumption of 100 orders per second. These blocks, which are integrated into OB 38 (300 ms), correspond to a communication load of 66%.

4.4 F-STOP

In the event of an F-STOP, either the entire F program (full shutdown) or just the F-shutdown group in which the error occurred (partial shutdown) is shut down. All F-runtime groups in an F-shutdown group are shut down at the same time. The F CPU's standard user program will continue to run in the event of an F-STOP.

When F shutdown groups are shut down:

- The outputs of the F signal modules controlled by the F-shutdown group are passivated.
- With S7 F Systems Lib V1_3 and higher, the F channel drivers of the F-shutdown group set the QBAD outputs to "1" and QUALITY to "0".
- Safety-related communication between the F-shutdown group and other F CPUs is interrupted.
- Data exchange between the F shutdown group and other F shutdown groups is interrupted.
- Where data exchange between the safety program and the standard user program is concerned, the standard user program is supplied with the last valid values.
- Block F_SHUTDN generates messages which are displayed automatically on the PCS 7 OS. In the case of S7 F Systems Lib V1_3, the messages contain the following text:
 - Safety program: Partial shutdown
 - Safety program: Complete shutdown
- The corresponding diagnostics events are written to the F CPU's diagnostics buffer.

4.4.1 Complete shutdown

All of the F-CPU's F-shutdown groups are shut down. Shutdown proceeds in the following order:

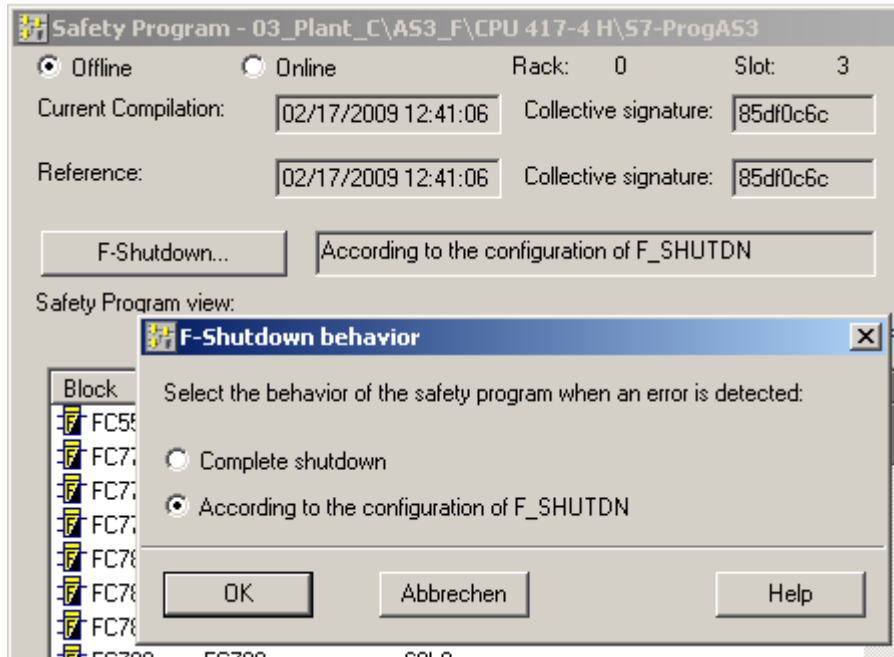
- First, the F-shutdown group in which the error was detected is shut down.
- All other F-shutdown groups are then shut down within double the time period you set as the F monitoring time for the slowest cyclic interrupt OB.

4.4.2 Partial shutdown

Only the F-shutdown group in which the error was detected is shut down.

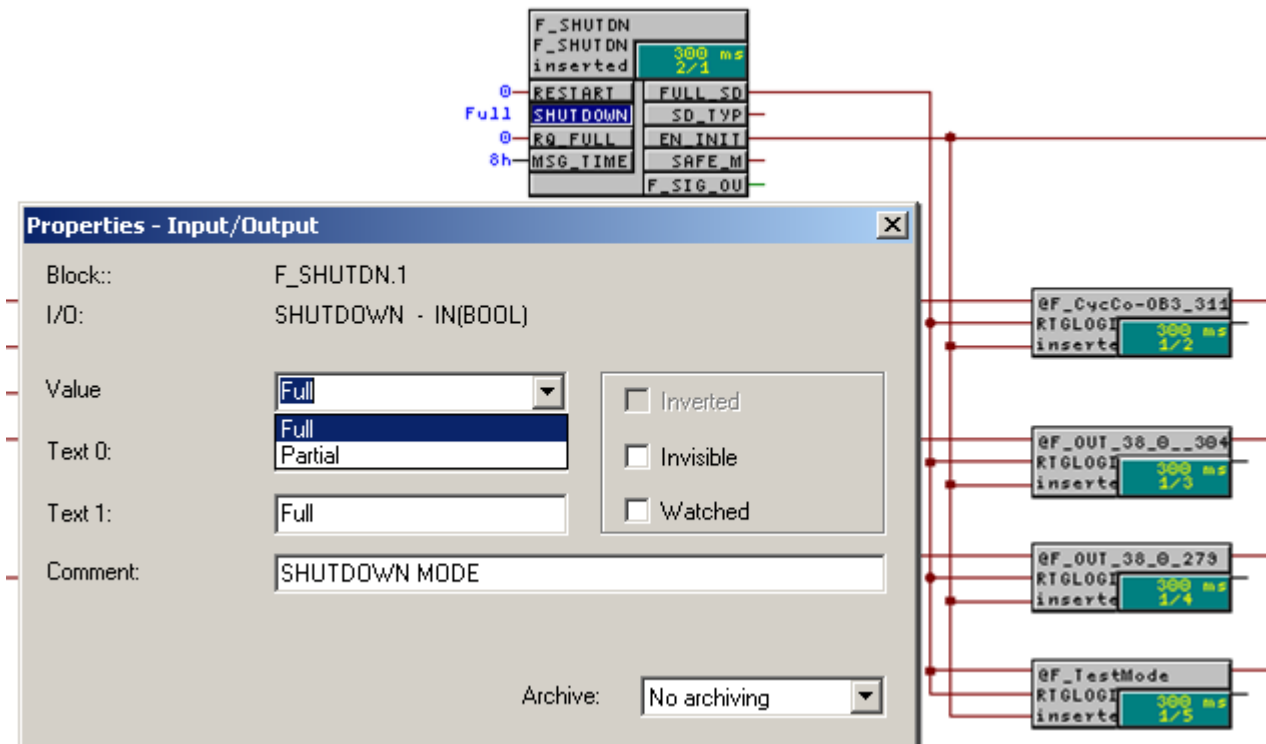
4.4.3 Parameter assignment for shutdown behavior

In the case of S7 F Systems V6.0 with S7 F Systems Lib V1_3, the shutdown behavior in the event of an F-STOP is defined in the "Safety Program" dialog, using the "Shutdown behavior" button.



You can use the "Shutdown behavior" dialog to select how the safety program should behave when an error is detected (in other words, in the event of an F-STOP):

- "Complete shutdown":
All F-shutdown groups associated with a safety program are shut down the first time an error is detected in an F-shutdown group.
- "Acc. to parameter assignment at F_SHUTDN":
Block F_SHUTDN is located in the @F_ShutDn chart. At the SHUTDOWN input, you can choose from:
 - "Partial":
The faulty F-shutdown group(s) is (are) shut down the first time an error is detected in an F-shutdown group (partial shutdown).
or
 - "Full":
All F shutdown groups associated with a safety program are shut down the first time an error is detected in an F shutdown group.



If you change the shutdown behavior, you must recompile the F program. This applies even if you have changed the shutdown behavior online in CFC.

4.4.4 Causes of errors

Errors that trigger an F-STOP

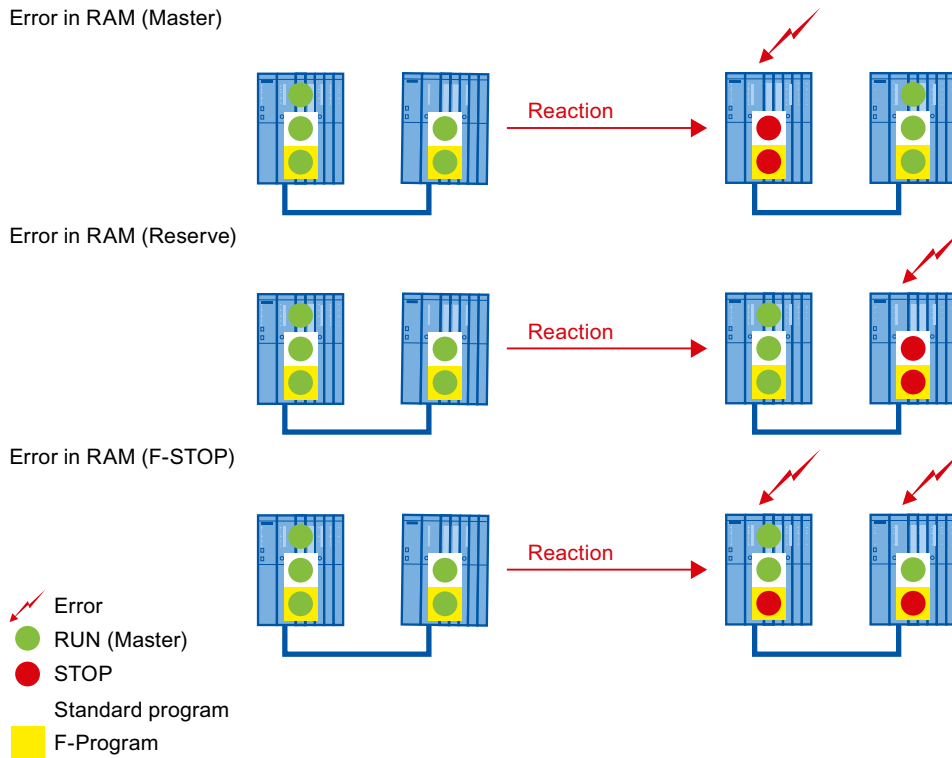
- Distortion of:
 - Data
 - Program sequence
 - Code
- CPU error

Errors that always trigger an F-STOP with full shutdown

Irrespective of the parameter assignment for F-STOP, a full shutdown is always triggered in the event of a cyclic interrupt OB request error (caused by a CPU/OB overload, for example).

4.4.5 Execution of an F-STOP in S7 F/FH systems

F-STOP illustration



Error in master:

Before a safety program in a redundant F CPU goes into F-STOP, it completes the following steps:

- The S7 F/FH system performs a master-to-standby switchover.
- The previous master goes into the configured operating state (default setting: TROUBLESHOOTING).

If no errors are detected, the F CPU reconnects.

Note

You can find more information in the manual titled *"Fault-tolerant Systems S7-400H"*. (<http://support.automation.siemens.com/WW/view/en/1186523>)

If an error is detected, the previous master goes into FAULT mode (all LEDs on the affected CPU flash).

On redundant F CPUs, errors on one communications partner will not stop program execution.

Error in both F CPUs:

The safety program goes into F-STOP immediately.

4.4.6 Exiting an F-STOP

Run an F-startup as described in the following chapter.

4.5 F startup and (re)start protection

4.5.1 F-startup

S7 F Systems does not make a distinction between a CPU cold restart and a CPU warm restart. The F_CHG_BO, F_CHG_R (part of the Safety Data Write function), and F_MOV_R (S7 F Systems Lib V1_3 and higher) F blocks are exceptions to this rule.

Both a CPU cold restart and a CPU warm restart will generate an F startup. In the event of an F startup the safety program starts up automatically with the initial values.

An F startup is performed:

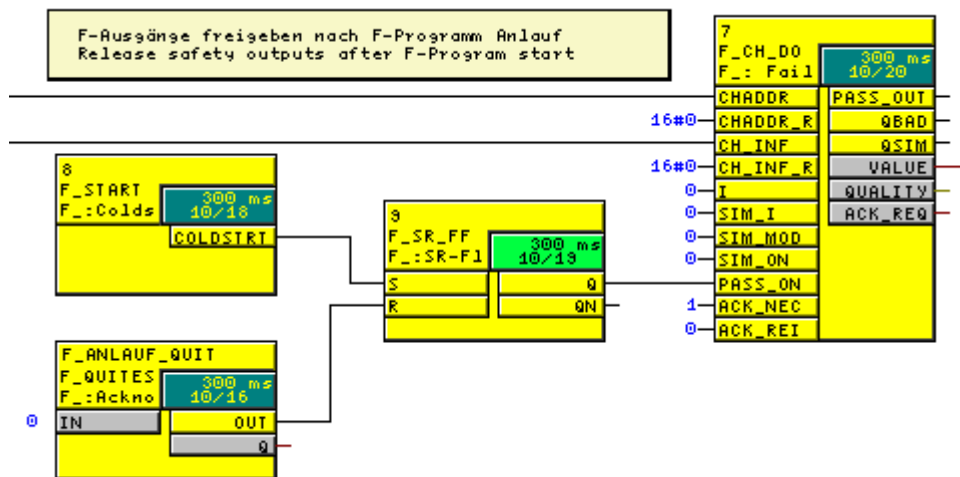
- After a CPU STOP, if you perform an F CPU warm restart
- After an F-STOP, if the "F_SHUTDOWN" F block detects a positive edge on the RESTART input.

Following a partial shutdown of the safety program, only the F-shutdown groups involved in the F-STOP perform an F startup. F-shutdown groups with errors remain in F-STOP.

4.5.2 (Re)start protection

If the process does not permit the safety program to start up automatically with the initial values, you will need to program a response to F-startup.

The F_START F-block is used to signal an F-startup of the safety program with the initial values. The COLDSTRT output parameter tells you that an F-startup has been triggered.



In this example, a flip-flop is set when the F program is started up, which passivates all the output drivers. Once all the process conditions have been fulfilled, the operator can release the F output signals by resetting the flip-flop using the F acknowledgment function (F_QUIRES block).

Note

For more options when programming (re)start protection, please refer to the "S7 F/FH Systems Configuring and Programming" Manual.

(<http://support.automation.siemens.com/WW/view/en/2201072>)

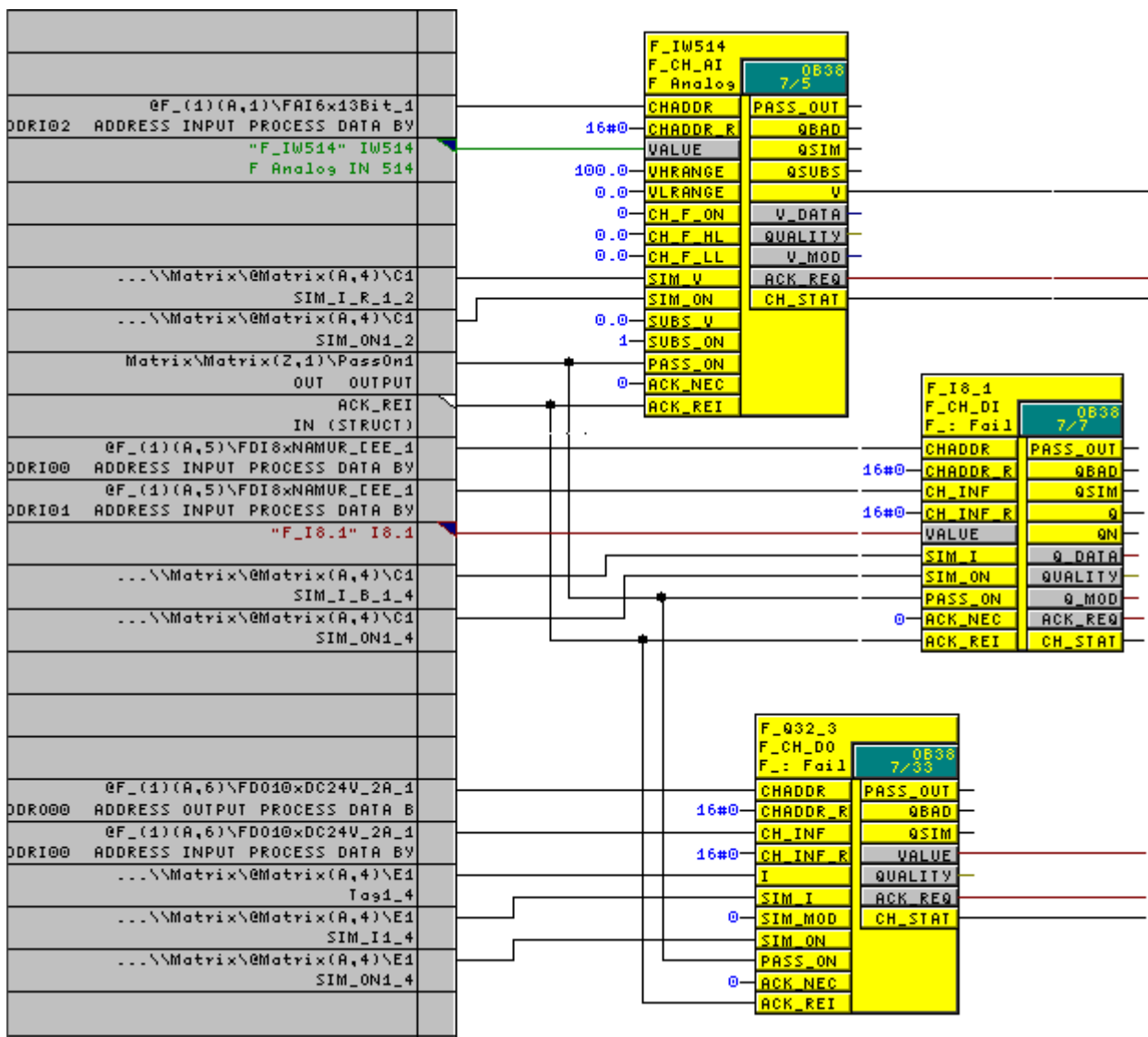
4.6 I/O access via F driver blocks

In S7 F Systems, F signal modules are accessed via F driver blocks rather than the process image. For this purpose, the following driver blocks are used in the program:

- F channel driver (e.g. F_CH_xx) for access to the input/output channels of F signal modules.

One F channel driver is required for every input or output channel used. Only one F channel driver is required for redundant channels.

In your safety program, F channel drivers form the interface to a channel of an F signal module and perform signal processing. F channel drivers vary depending on the F signal modules. They are placed and interconnected in the safety program by the user.



- The CFC driver wizard creates and interconnects one F module driver per module for PROFIsafe communication purposes.

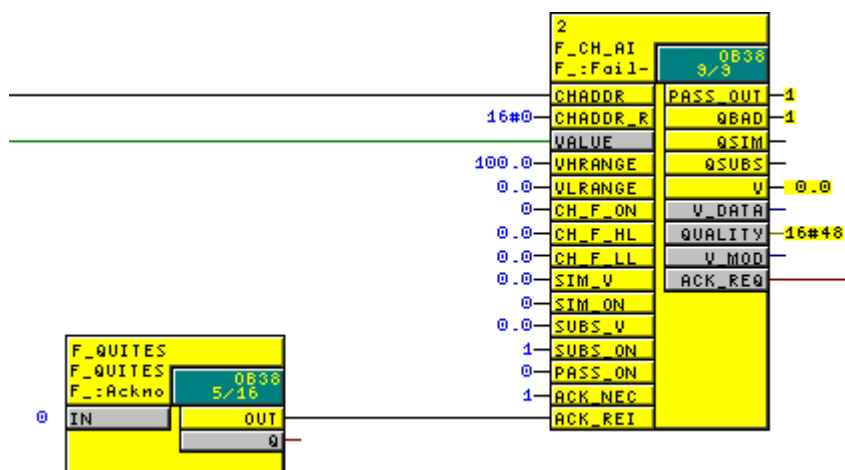
4.7 Passivation and reintegration of input/output modules

4.7.1 Passivation - general

Passivation means that in the event of an error, one or a number of channels on an F signal module are switched to a safe state. In the event of a channel error (a faulty sensor, for example), only the affected channel is passivated.

In the event of a module error (a communication error, for example), all channels on the fail-safe I/O module are passivated.

If an F signal module detects an error, it switches the affected channel or all of its channels to the safe state; in other words, the channels on this module are passivated. The fail-safe F signal module sends a message to the F channel driver and the PCS 7 OS to indicate that it has detected an error.



The PASS_ON input can also be used to activate and deactivate the passivation of a channel in the safety program, e.g. by using a specific condition in the program sequence or restart protection.

When output channels are passivated, the outputs are de-energized (set to a zero-current or zero-voltage state). The F channel driver of a passivated digital output channel issues a substitute value with the quality code (QUALITY) 16#48 and the QBAD output is set to 1.

When input channels are passivated, substitute values are forwarded to the safety program, regardless of the actual process signal.

The F channel driver of a passivated digital input channel outputs substitute value 0 with quality code (QUALITY) 16#48 and the QBAD output is set to 1. In accordance with the parameter assignment at the SUBS_ON input, the F channel driver of an analog input channel will output either a substitute value with quality code (QUALITY) 16#48 or the last valid value with quality code (QUALITY) 16#44. The QBAD output is also set to 1 and, if a substitute value is output, the QSUBS output is set to 1.

See also

F startup and (re)start protection (Page 81)

4.7.2 Group passivation

If, during passivation of an F I/O or a channel of an F I/O, you wish to activate passivation of other F I/Os, you can use the PASS_OUT output/PASS_ON input to perform group passivation of related F I/Os.

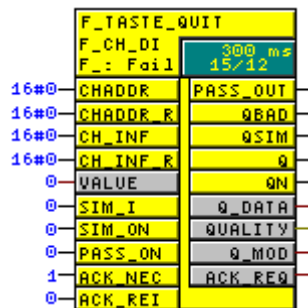
Group passivation via PASS_OUT/PASS_ON can, for example, be used to force simultaneous reintegration of all F I/Os after the F system has been started up.

To enable group passivation, you must link all PASS_OUT outputs of the F channel drivers in this group with F_OR4 F blocks, and interconnect the OUT output result of F_OR4 with all PASS_ON inputs of the F channel drivers in this group.

4.7.3 Reintegration following elimination of errors

Reintegration means:

- Valid process values start to be output again at the output channels of the fail-safe output modules.
- The F-channel drivers associated with the fail-safe input modules resume the forwarding of valid process values to the safety program.
- Once a channel error has been eliminated, a channel on a fail-safe module can be reintegrated automatically or following user acknowledgment. You can use the ACK_NEC input of an F-channel driver to specify whether or not user acknowledgment is required:
 - Value 0: Automatic reintegration without user acknowledgment
 - Value 1: Prompt for user acknowledgment for reintegration following error elimination



If passivation was triggered by setting PASS_ON = 1, user acknowledgment will not be required for reintegration.

Note

The ACK_NEC input can only be set to 0 if the process permits automatic reintegration from a safety-related point of view.

4.7.4 Automatic reintegration on channel error

If the ACK_NEC input is not set, once the channel error has been eliminated (with the exception of PROFIsafe communication errors and module errors) the affected channel will be reintegrated (depassivated) automatically as follows:

- With input modules - immediately
- With output modules - within a matter of minutes (due to required test signal injections, after 2 successful test cycles).

Note

User acknowledgment is always required for reintegration following module errors (ACK_REQ output set), even if ACK_NEC has not been set. An interconnection of the ACK_REI input with an automatically generated signal is not permitted.

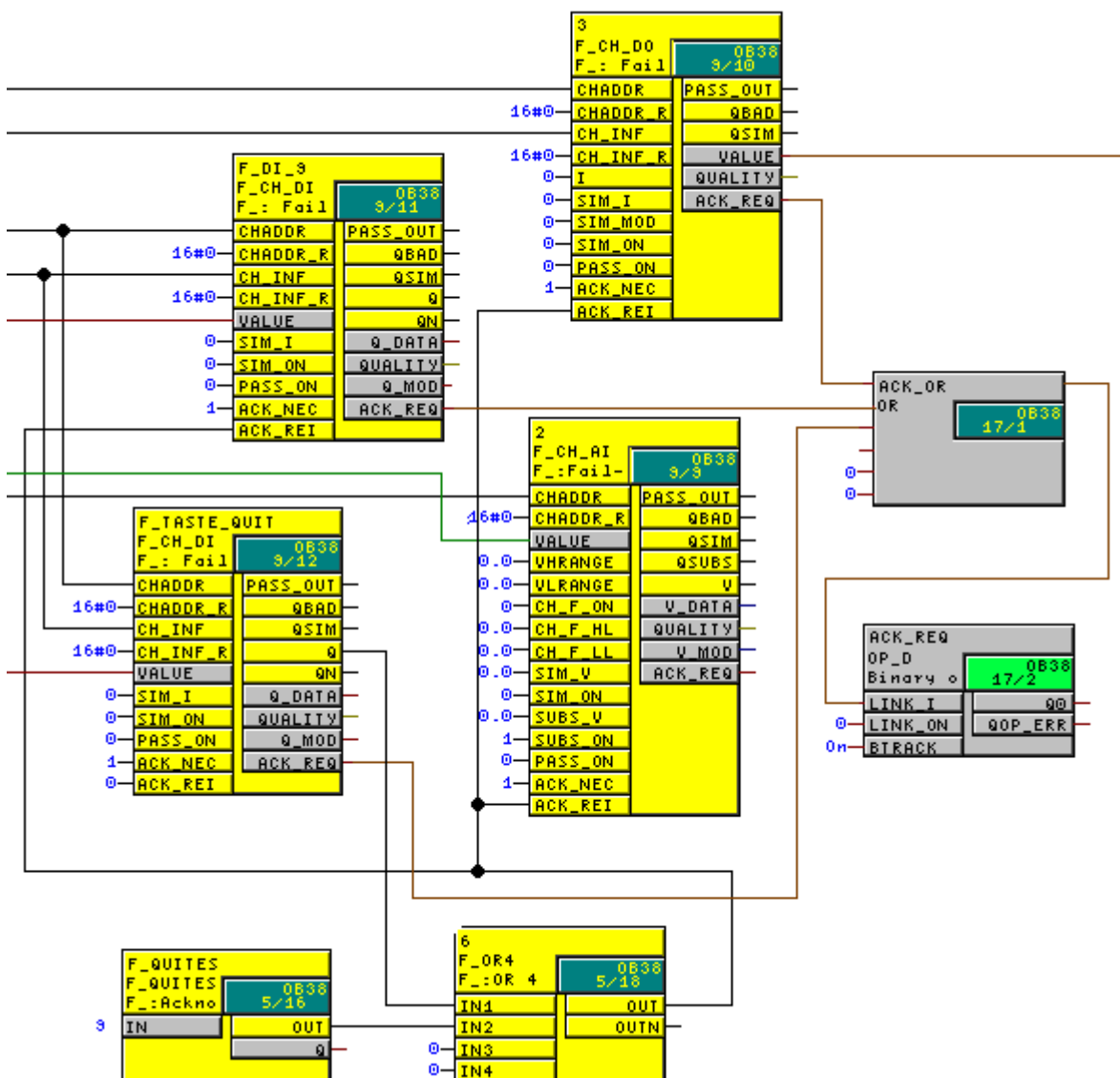
4.7.5 Programming reintegration following user acknowledgment

A value of 1 at the ACK_REQ output of the F channel driver indicates that the error has been eliminated and user acknowledgment for reintegration is possible.

Setting the ACK_NEC input will delay reintegration of the input or output channel until after user acknowledgment with a positive edge at the F channel driver's ACK_REI input.

Options for user acknowledgment

- Connection of an acknowledgment key to a fail-safe digital input module
- Manual input from an ES/OS station using the F_QUITES block



If you use an acknowledgment key for user acknowledgment, in the event of a module error on the F signal module group to which the acknowledgment button has been connected, you will no longer be able to trigger acknowledgment to reintegrate this F signal module group. This "block" can only be lifted by means of an F CPU STOP/RUN transition.

Therefore, in cases where you are setting up acknowledgment for reintegration of an F signal module group to which an acknowledgment key is connected, we recommend providing another means of acknowledgment via an OS.

NOTICE
Automatic user acknowledgment is not permitted.

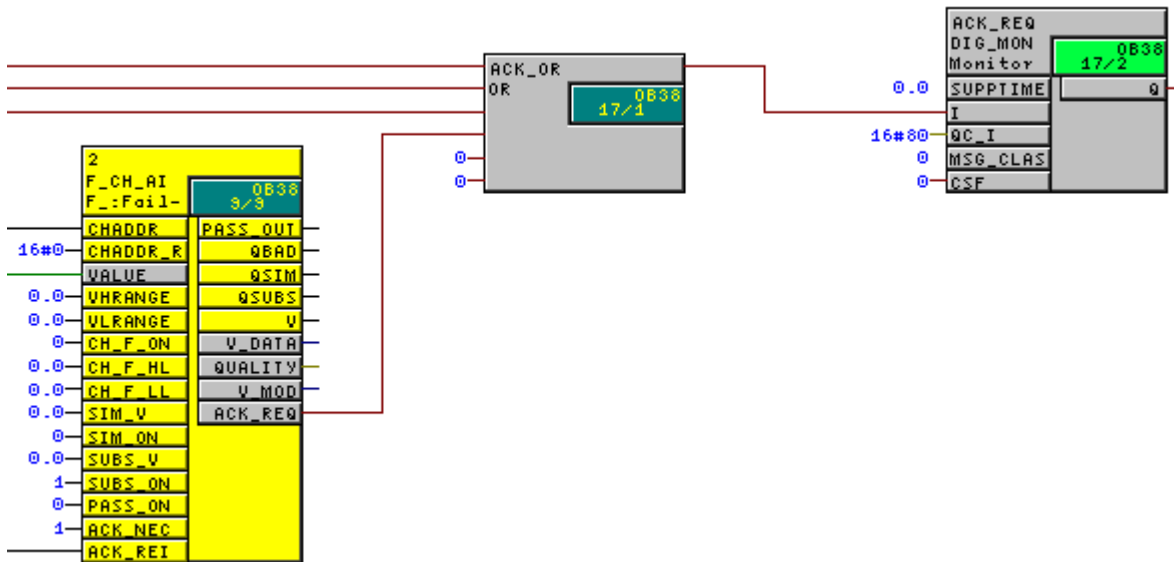
How to program user acknowledgment via an OS

1. Add the F_QUITES F block to your safety program. You can access the acknowledgment signal for evaluation for user acknowledgments at the output OUT of F_QUITES.
2. Interconnect the OUT output of F_QUITES with the ACK_REI input of the F channel drivers.
3. On your OS, set up a button for writing the "Acknowledge value" "6" (first acknowledgment step) and a second button for writing "Acknowledge value" "9" (second acknowledgment step). Connect the buttons to the IN input of F_QUITES.
4. Optional: On your OS, evaluate output Q of F_QUITES to show the time window within which the second acknowledgment step must be completed (can be viewed by activating the second button) or to show that the first acknowledgment step has already been completed.

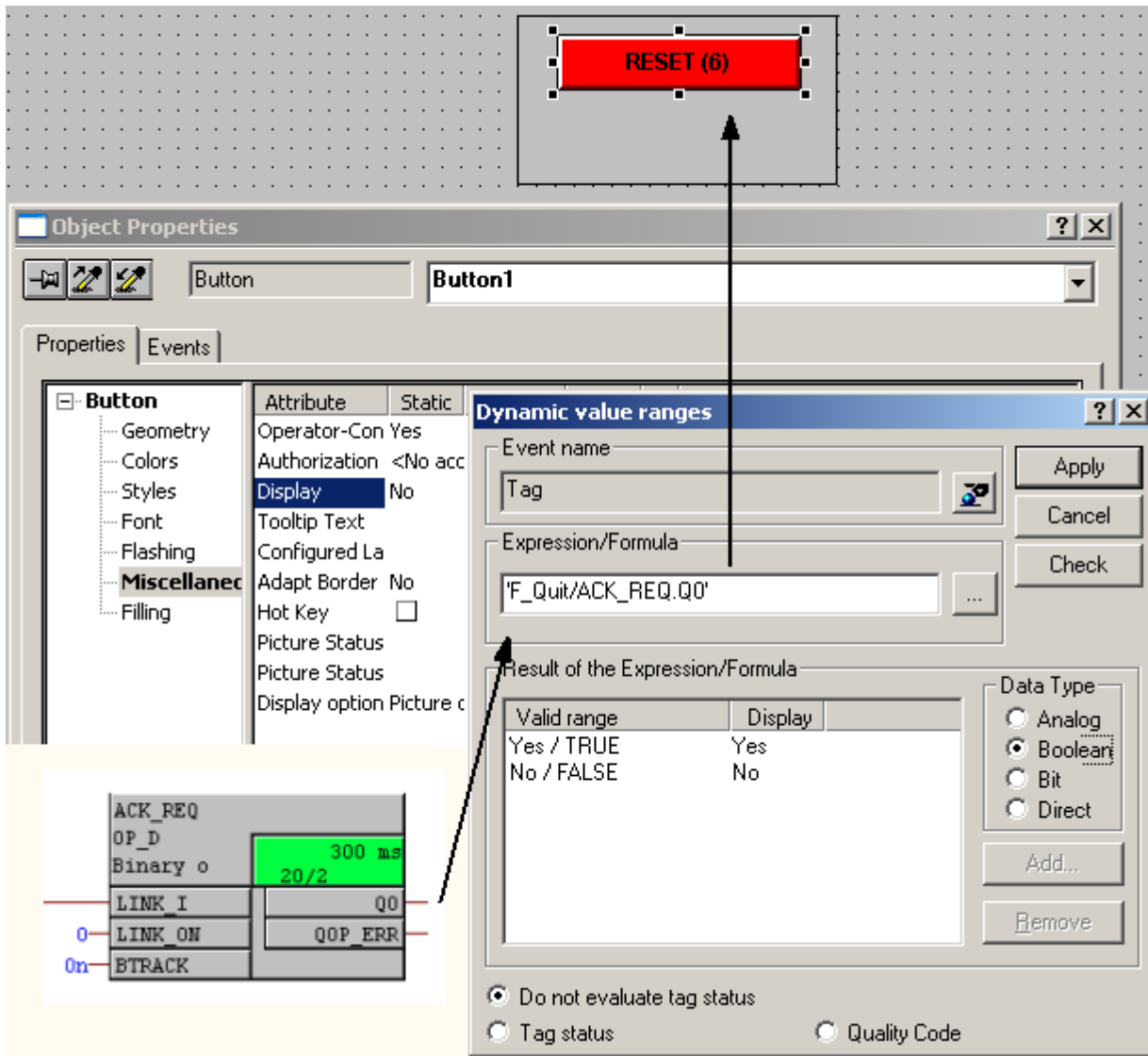
NOTICE
Automatic reintegration using F_QUITES: The non-safety-related input IN of F_QUITES must not be interconnected with a signal or described by a signal which generates the above condition (change from 6 to 9 within a minute) automatically for a fail-safe acknowledgment. Fail-safe acknowledgment must only be generated by means of a conscious manual entry on the ES/OS (not automatically in the program).

Example: Implementing F user acknowledgment in the OS

All "ACK_REQ" channel driver outputs are grouped by means of an OR in the standard user program and made available to the OS via a DIG_MON block.

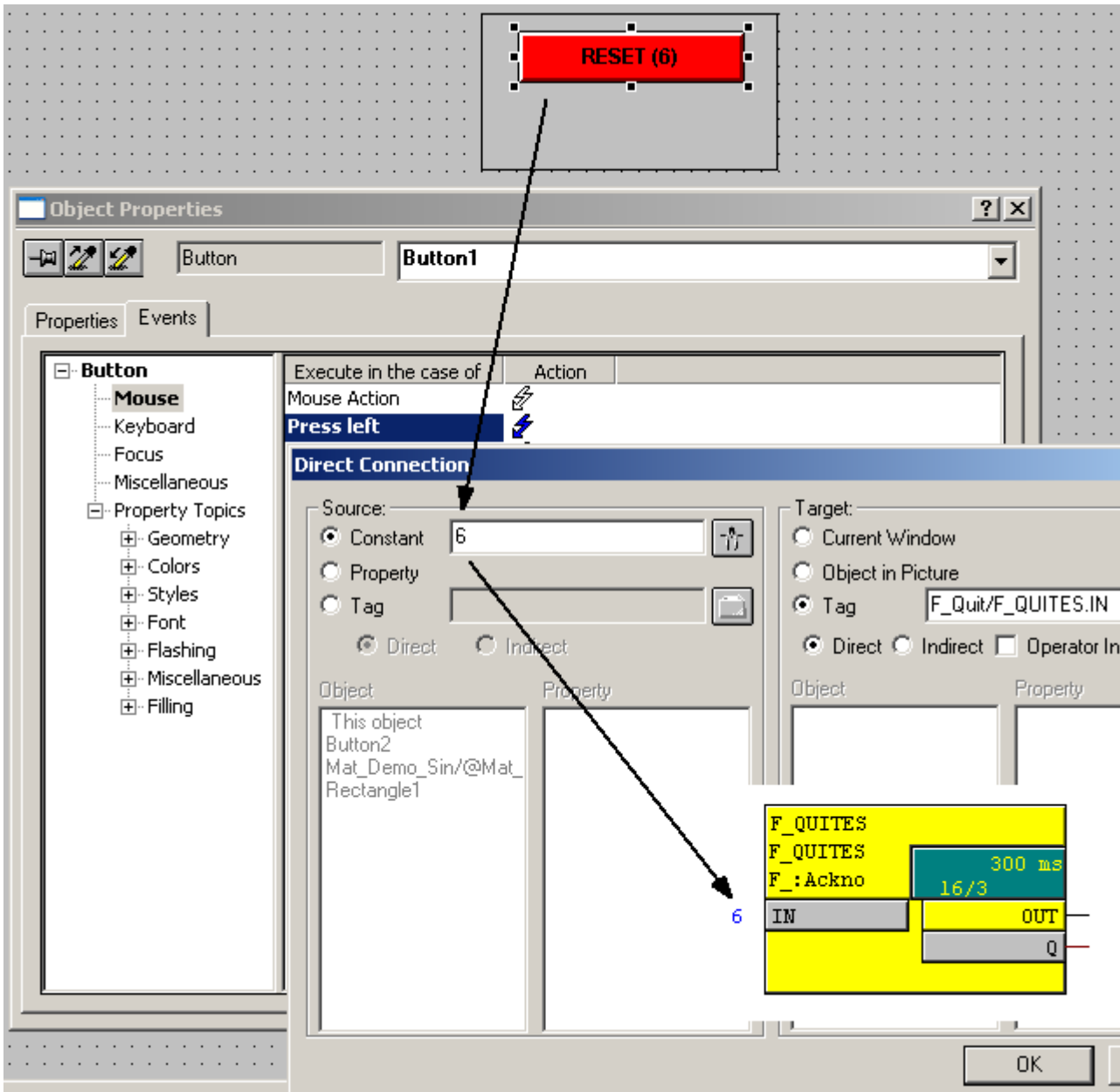


If an acknowledgment prompt is pending (ACK_REQ=1) the acknowledge field (yellow) and the "reset (6)" button will appear on the OS.

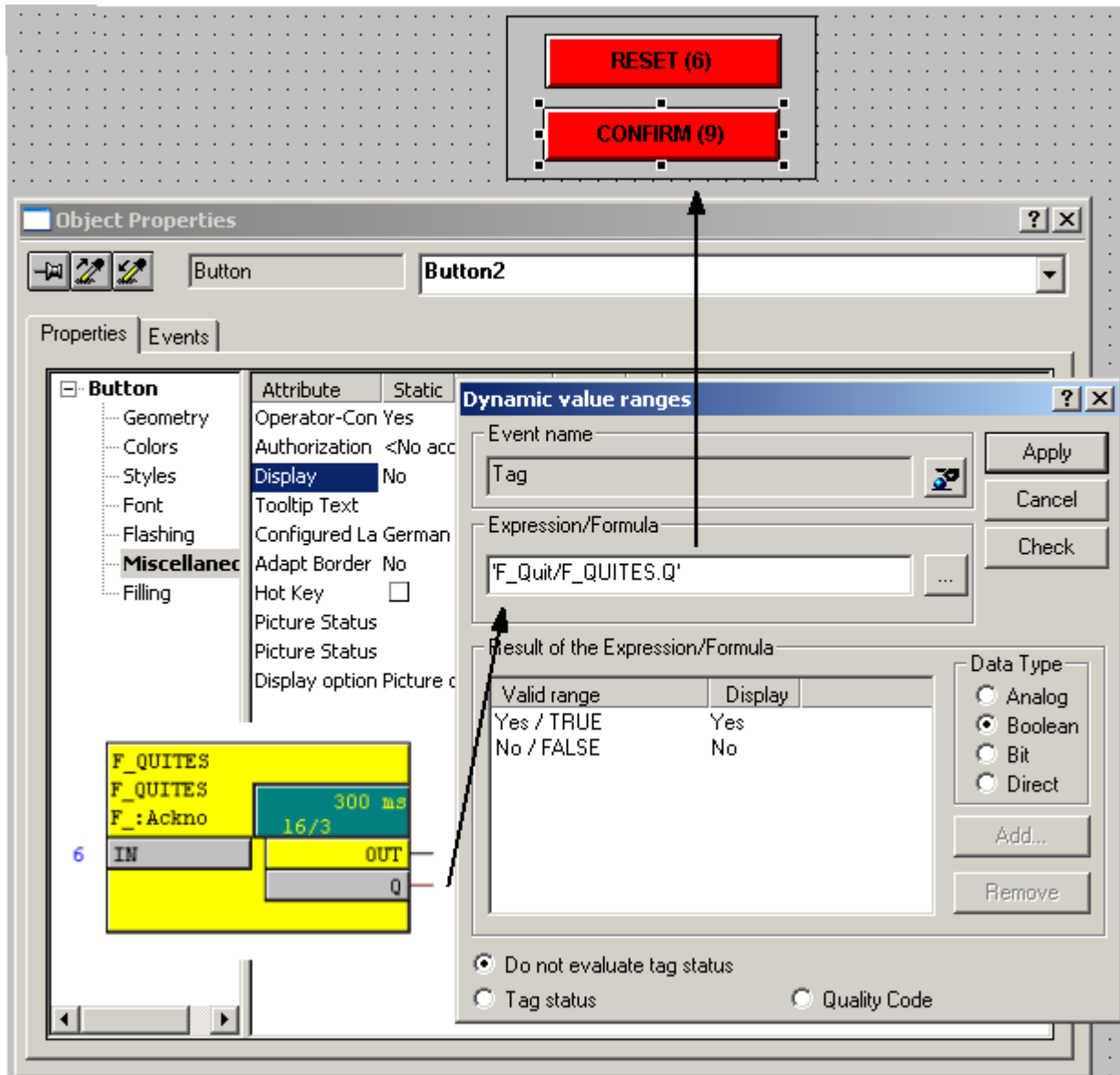


Procedure

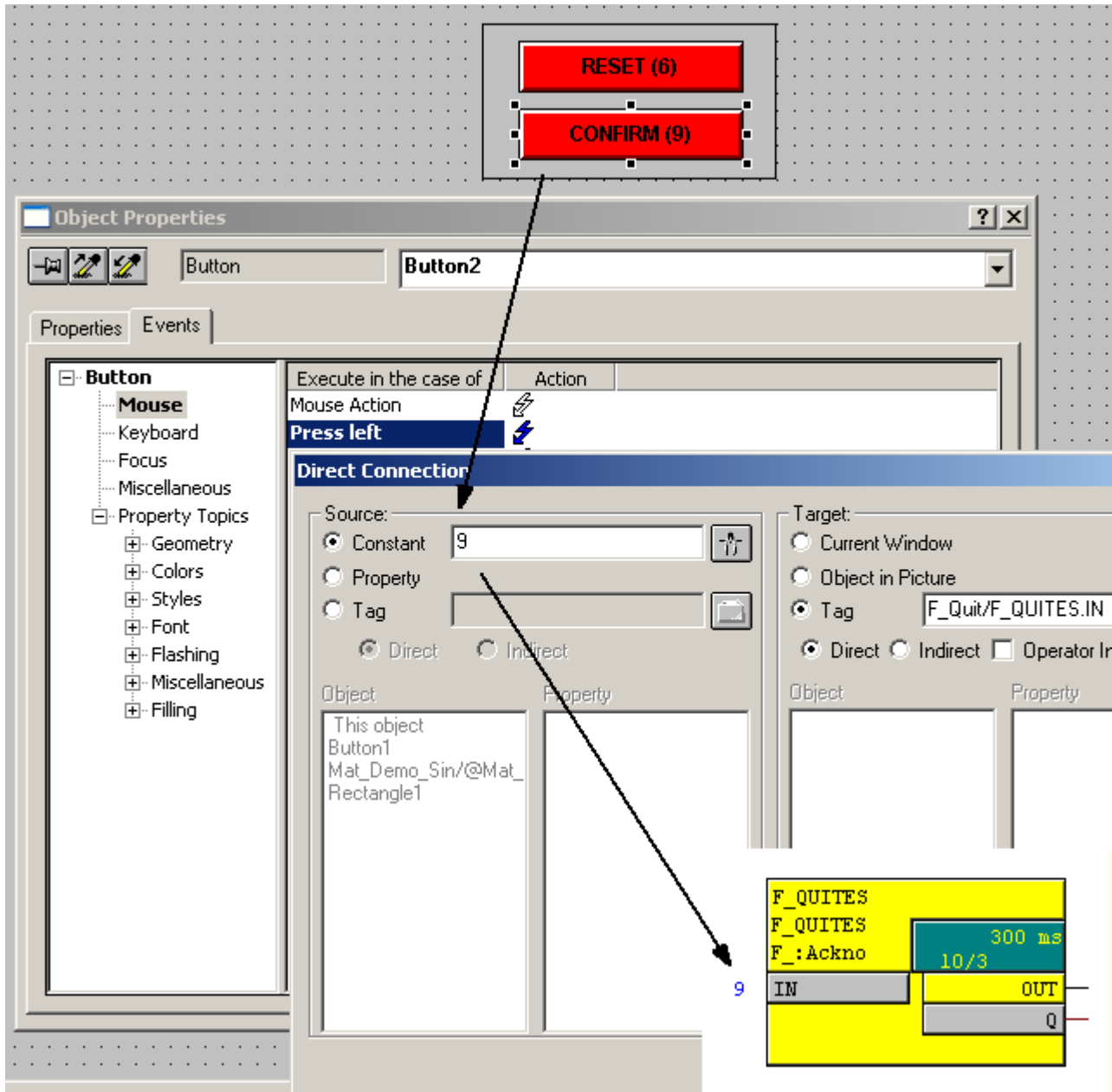
1. Press the first acknowledge button, "reset (6)", to write the value 6 to the "IN" input of the F_QUITES block.



The second acknowledge button, "reset (9)", appears if the "Q" output of the F_QUITES block has been set. This output remains set for 60 seconds.



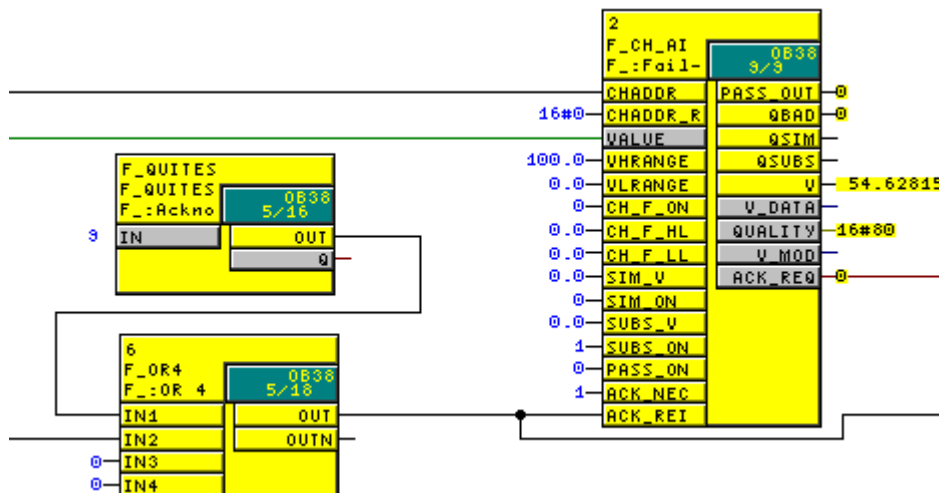
2. Press the second acknowledge button, "reset (9)", to write the value 9 to the "IN" input of the F_QUITES block.



The "OUT" output of F_QUITES is set to 1 for one cycle and the F channel drivers are reintegrated.

Result

If the value 6 is written to the "IN" input of the F_QUITES block, followed by the value 9 within 60 seconds, the "OUT" output of F_QUITES will be set to 1 for one cycle. The F channel drivers connected to this output (at the "ACK_REI" input) are reintegrated if they are ready for acknowledgment (ACK_REQ = 1).



If the acknowledgment prompt is no longer pending, the buttons on the OS are hidden again.

4.8 Compiling the F-program

If an S7 program contains charts with F blocks, these will be compiled when the CFC charts are compiled. Measures for eliminating errors will also be expanded and additional safety-relevant checks carried out.

4.8.1 Password protection for the safety program

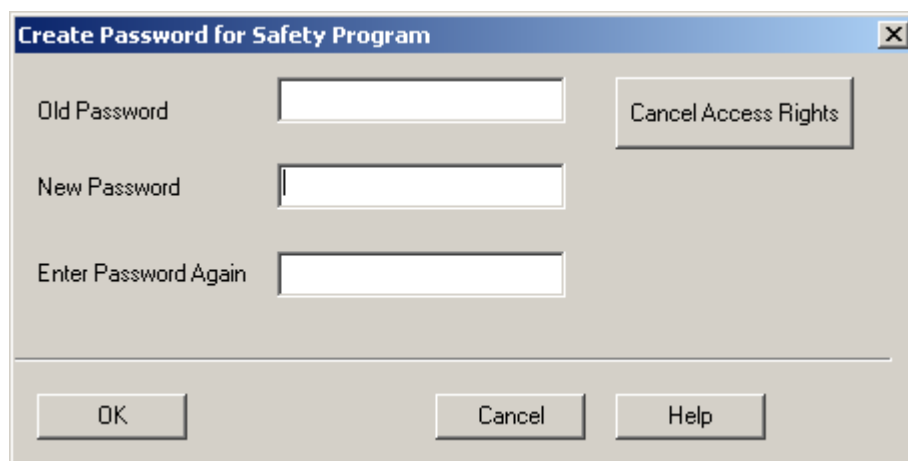
A password protects the fail-safe program against unauthorized compilation. The password for compiling the CFC charts is requested the first time the system carries out compilation.

If the password is entered, it remains valid for one hour. If you have finished your work, reset the password's period of validity.

To do this, in the SIMATIC Manager select the CPU that contains the safety program and use the "Options > Edit Safety Program" menu command to open the relevant dialog.

Here, you will find the "Password" button that opens the "Create Password for Safety Program" window. You can use the "Cancel Access Rights" button to prevent unauthorized compilation of the safety program.

You also have the option of changing the password.

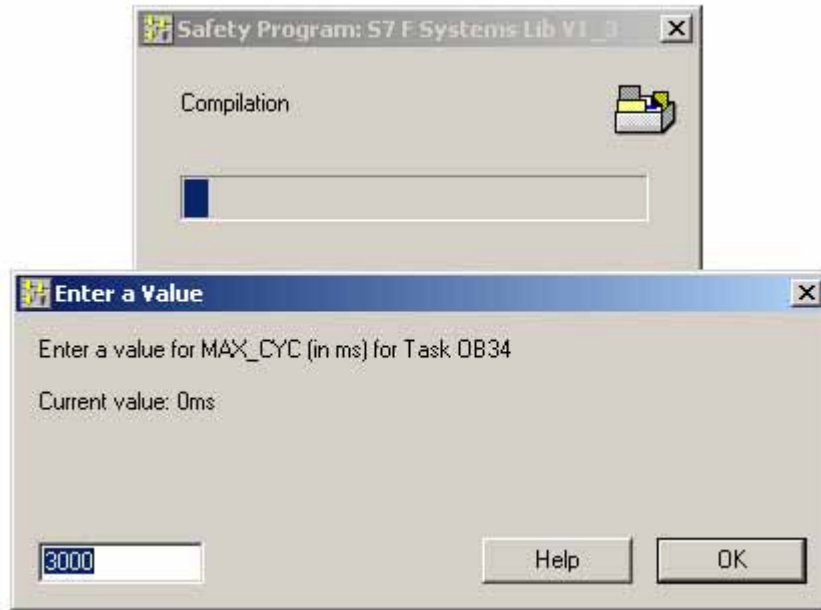


If changes to the fail-safe program part are detected during compilation, you will need to enter the password for the safety program.

If a standard user program and a fail-safe program are running in one CPU, changes to the standard part can be compiled without the need to enter the F password.

4.8.2 Parameterizing the maximum F cycle monitoring time

The F CPU runs F cycle time monitoring for every cyclic interrupt OB containing F-runtime groups. The first time the F program is compiled, for each cyclic interrupt OB which contains an F program you will be prompted to enter a value for the maximum cycle time (MAX_CYC) that may elapse between two calls of this cyclic interrupt OB. MAX_CYC denotes the maximum time that may elapse between two calls of this cyclic interrupt OB.



The default for the maximum F cycle time is 3,000 milliseconds.

Check whether this setting is suitable for your process. Change the default if necessary.

Note

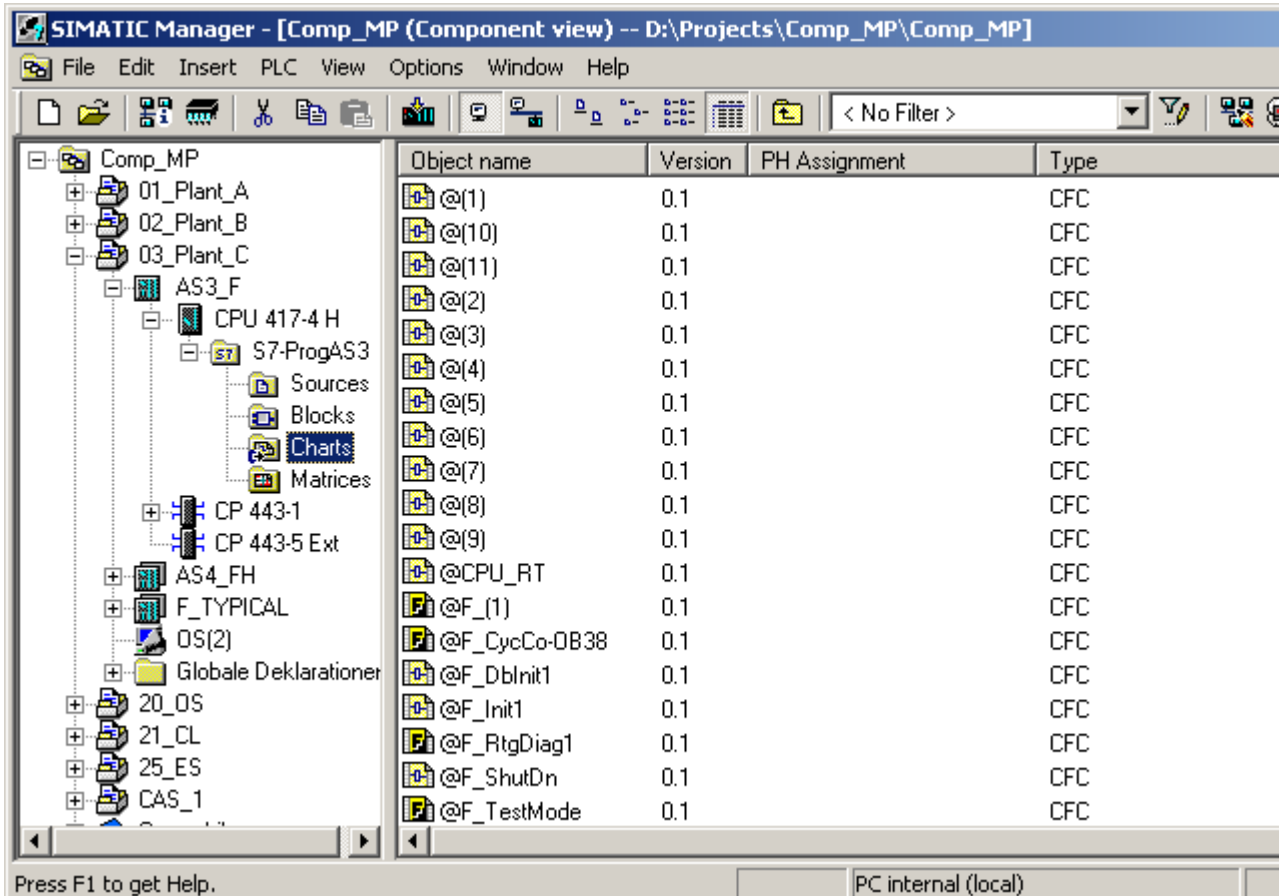
You can change the default value at the MAX_CYC input of the F_CYC_CO block in chart @F_CycCo-OB3x whenever you wish.

You will find information about setting the F monitoring time and reaction times in Section Monitoring times and system response times (Page 110).

4.8.3 Compiling the S7 program

During compilation, the S7 program is automatically expanded to include diagnostics drivers (contained in the @ system charts) and F-specific parts.

F system blocks are stored in @F_xxxx charts.

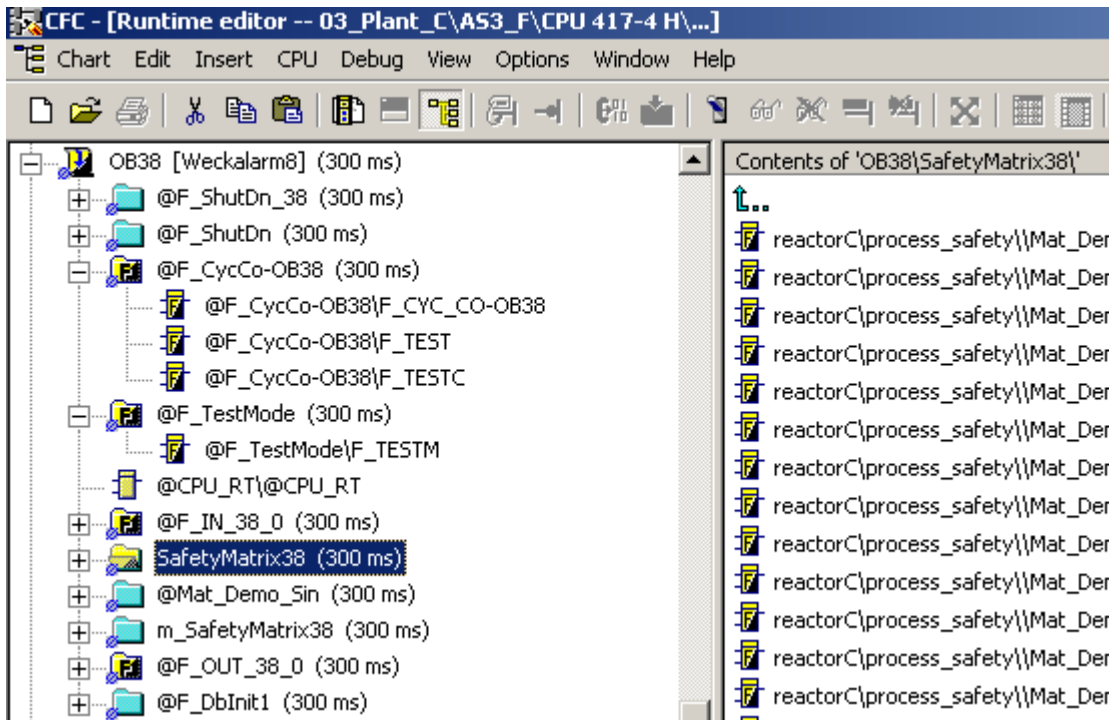


Note

Placements, interconnections, and parameter assignments for F-system blocks completed automatically during the compilation process must not be changed.

You must not change or delete F blocks in the block container.

The CFC compiler also automatically places F system blocks needed for the operation of the safety program in runtime groups. The names of these runtime groups begin with "@F_".



Note

The CFC charts and runtime groups with fail-safe blocks appear in yellow and are marked "F" to distinguish them from standard charts.

4.9 Safety mode and downloading the safety program

Safety mode of the safety program in the F-CPU can be temporarily deactivated and reactivated. This enables you to make changes to the safety program in RUN mode.

4.9.1 Information on safety mode

An S7-400 F/FH system containing a fail-safe program automatically goes into safety mode when it starts up. In safety mode, all functions present in the system for system error detection and for the fail-safe user program are activated. In this state it is not possible to modify the safety program in active operation (RUN).

In order to make online changes to fail-safe parameters from the CFC online, or to download changes in the fail-safe program, part of the diagnostics functions must be switched off.

Safety mode must be deactivated for this purpose, before changes are made online or downloaded.

Prior to deactivating safety mode, you must ensure that the process is in a non-critical state and is being monitored by an operator during this time (monitored operation).

To download program changes in deactivated safety mode, the monitoring parts are switched off, the software changes are detected, and an F-STOP is triggered. "Random hardware faults" continue to be detected and the diagnostics for the modules remain active. The safety program continues to be processed to ensure that a "Demand" from the field leads to a safety shutdown.

Once the changes have been made, or at the end of the download process, safety mode must be reactivated immediately.

The risk analysis may reveal details of other measures that are required.

4.9.2 Deactivating safety mode

Safety mode can be deactivated/activated if a relevant system prompt appears, or from the SIMATIC Manager. To do this, select the CPU that contains the safety program and use the "Options > Edit Safety Program" menu command to open the relevant window.

The field underneath the "Safety Mode..." button shows you whether safety mode is "activated" or "deactivated". If the safety program does not match the safety program in the F CPU or communication with the F CPU has failed, "unknown" will appear here.

If there is a connection between the engineering station and the CPU, the current status of the safety program is displayed and can be changed using the "Safety Mode..." button. An additional prompt appears before the status is changed.

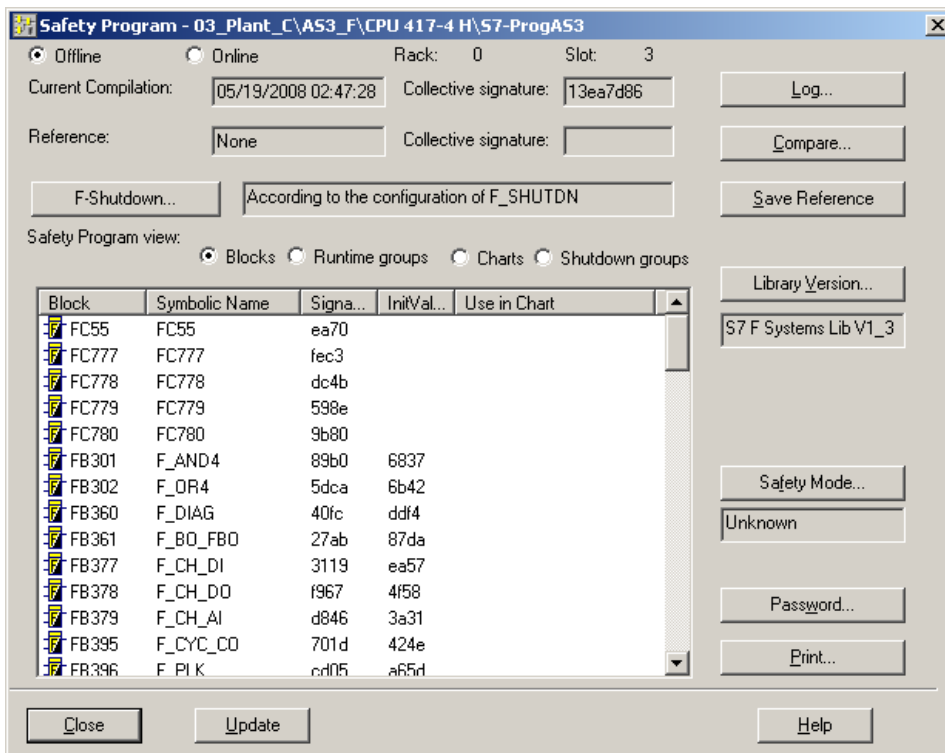
The safety mode status is entered in the CPU diagnostic buffer and reported on the OS, and can be checked in chart @F_Shutdn at the SAFE_M output of the F_SHUTDOWN block.

Preconditions for deactivating safety mode

- The CPU must be in the RUN state (mode switch in RUN or RUN-P).
- Safety mode must be activated.

Procedure

1. Select the CPU or its S7 program in the SIMATIC Manager.
2. Select the menu command "Options > Edit Safety Program".



3. Click the "Safety mode" button and (if applicable) enter the password for the safety program.
4. Confirm the deactivation of the safety mode.

Note

The F_SHUTDN block generates a message when safety mode is activated/deactivated. The parameter for the message repetition time is assigned at the F_SHUTDN block in chart @F_ShutDn.

Please note the following when deactivating safety mode

Manual intervention in the safety mode of fail-safe systems requires particular care and attention.

- Any changes must be made in accordance with current change management guidelines.
- An influence analysis must be carried out on any changes to be made to the active process.
- The changes must undergo a function test that complies with relevant standards (e.g. IEC 61511-1) before the program can be imported into the active plant.

As a general principle, the effect of changes on the system characteristics (program runtime, etc.) must be analyzed. Following this, a final function test of all systems affected by the changes (validation) must take place.

- The appropriate formal procedures for approval must be put in place before safety mode can be deactivated.

Notification of the plant personnel in charge, deactivation of safety mode, change downloading (delta downloading), and, finally, activation of safety mode must be accompanied by the relevant documentation.

- This documentation will then include the results of the change comparison and its analysis, the log of the relevant function test, and the updated hardware and software documents, together with modified signatures.

While safety mode is deactivated, we highly recommend operating the plant with a high level of supervision by the operating personnel. The measures required for this purpose are derived from the influence analysis. This is urgently recommended due to the concluding safety validation following a program change.

Safety mode should be reactivated as soon as the changes have been performed.

4.9.3 Activating safety mode

Following a download of changes, you will need to reactivate safety mode in order to ensure secure execution of the the safety program.

Procedure

1. Select the CPU or its S7 program in the SIMATIC Manager.
2. Select the menu command "Options > Edit Safety Program".
3. Click the "Safety mode" button.
4. Confirm activation of the safety mode.

Note

If, when safety mode is deactivated, the safety program detects a safety-related error, it will disable the option to activate safety mode. A corresponding message will appear indicating how you can rectify the problem.

4.9.4 Downloading the safety program

After compilation, you can download the program. Depending on whether safety mode is activated or deactivated, you can download the entire safety program or changes to it as follows:

Download	CPU in STOP	CPU in RUN safety mode active	CPU in RUN safety mode not active
... The entire program	Possible	Not possible	Not possible
... Changes in the standard program	Not possible	Possible	Possible
... Changes in the safety program	Not possible	Not possible	Possible

Requirements

- The station's hardware configuration data has been downloaded to the CPU.
- The user program has been compiled without errors.
- You have access rights to the CPU.
- There is an online connection between the CPU and your ES.

Rules for downloading

- Empty F-runtime groups may arise when deleting safety functions. To remove these with PCS 7 V7.0, you need to delete them before compilation takes place. To do this, in the CFC editor select "Edit > Delete Empty Runtime Groups". With PCS 7 V6.1, you need to activate the "Delete Empty Runtime Groups" option when carrying out compilation.
- You can only download the safety program from the CFC editor or the SIMATIC Manager via the chart folder.
- Once you have downloaded an acceptance-tested safety program, you will need to check the collective signature in the same way as during acceptance testing.

Procedure

To download the safety program, select the menu command "CPU > Download > Entire program" in the CFC editor. This will set the F CPU to STOP.

To download changes made to the safety program, select the menu command "CPU > Download > Changes" in the CFC editor. Depending on the CFC editor version you are working with, you may need to deactivate safety mode in the SIMATIC Manager before this process takes place and reactivate it afterwards. Alternatively, a prompt may be issued in CFC offering you the option of deactivating safety mode directly and reactivating it after downloading has taken place.

Note

Before the safety program is downloaded you will be prompted to enter the CPU password if changes are detected in the fail-safe part of the program.

Once you have downloaded the program to the CPU, you will need to compare this program's collective signature with the collective signature in the acceptance-tested printout. On S7 F/FH systems, you need to run this comparison for both CPUs.

See also

Tracking changes in the safety program (Page 144)

4.10 Working with safety-related parameters on a PCS 7 OS

Changes to fail-safe parameters on a PCS 7 OS can be made using the following options:

- Safety Data Write (SDW)
- Safety Matrix
- F_QUITES for fail-safe acknowledgment

4.10.1 Safety Data Write (SDW)

Introduction

The "Safety Data Write" function enables safety-related changes to be made to F parameters in the safety program of an F CPU via an OS.

A special safety protocol is used to make changes to F parameters in safety mode. This meets Safety Integrity Level requirements up to SIL3 in accordance with IEC 61508. Modified values of F parameters can also be retained following a restart (warm restart) of F/FH systems.

The optional S7 F Systems software offers the following for SDW:

- Two F blocks, which you integrate into the CFC charts in your safety program
 - F_CHG_R: SDW for F_REAL data type F parameters
 - F_CHG_BO: SDW for F_BOOL data type F parameters
- Associated faceplates, which you integrate into your OS. Two identification values (Safe_ID1 and Safe_ID2) are used to uniquely assign each faceplate to a block in an AS.

Transaction for SDW

You can use SDW to modify an F parameter in the safety program of an F CPU if you execute a specific sequence of operations on the OS within a specific period of time. The entire change operation is known as a "transaction".

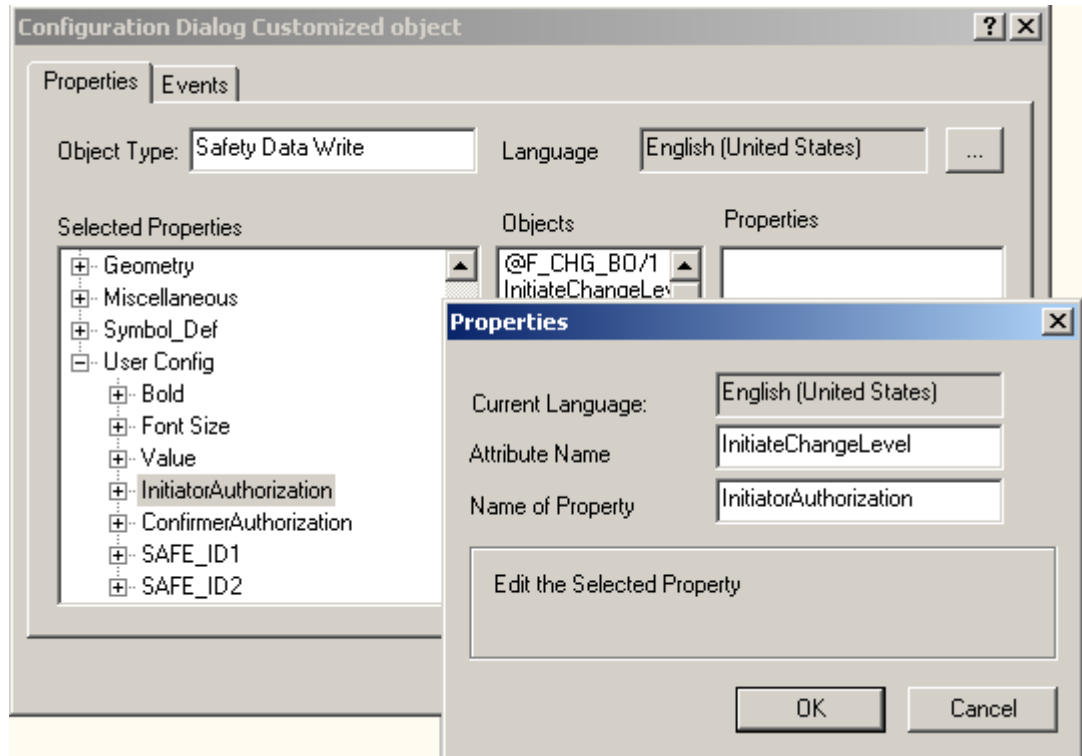
Operator types for SDW

A transaction can only be performed by an individual operator who initiates, checks, and confirms the change. However, one transaction can be performed by two operators. The first operator initiates the change (initiator) and the second re-enters, checks, and confirms the value (confirmer).

Operator authorizations and Safe_IDs are set in the faceplate properties. They are overwritten with the default settings whenever the faceplates are updated, however. To prevent this, enter the parameters in file @@PCS7Typicals.CFG,

which you can find in the project path of the OS server project on the ES, under wincproj\<OSname>\WScripts.

The names of these attributes and properties can be found in the configuration dialog of the block icon, by selecting the block icon followed by "Configuration dialog..." in the context menu.



In file @@PCS7Typicals.CFG, you will need to extend the [Columns] and [Column00] sections. The name of the property is entered in [Columns] and the attribute in [Column00]. The sections should then appear as shown below (please note that they are case-sensitive).

Since the sections already contain entries for properties and attributes, you need to proceed to the last line in each case. Start with the first unused number (in the example: last number used = 10, first unused number = 11 = N), and increase the number for each entry.

```
[Columns]
...(existing entries)
ColumnN = SAFE_ID1
TypeN = 3
ColumnN+1 = SAFE_ID2
TypeN+1 = 3
ColumnN+2 = InitiatorAuthorization
TypeN+2 = 3
ColumnN+3 = ConfirmerAuthorization
TypeN+3 = 3
```

```
[Column00]
...(existing entries)
[ColumnN]
Property0 = SAFE_ID1
[ColumnN+1]
```

4.10 Working with safety-related parameters on a PCS 7 OS

Property0 = SAFE_ID2
[ColumnN+2]
Property0 = InitiatorAuthorization
[ColumnN+3]
Property0 = ConfirmerAuthorization

Note

For more details of SDW, please refer to the "S7 F/FH Systems Configuring and Programming" Manual. (<http://support.automation.siemens.com/WW/view/en/2201072>)

For more information on the structure of the @@PCS7Typicals.CFG file, refer to the WinCC Information System (Start > Simatic > WinCC > WinCC Information System in contents under Options > Options for Process Control > Graphic Object Update Wizard > Structure of the Configuration File).

4.10.2 F_QUITES

Using the "F_QUITES" F block, the OS can generate fail-safe pulses in the F program of the automation system. You can find an application example for F_QUITES in "Implementing F user acknowledgment in the OS", in Section Programming reintegration following user acknowledgment (Page 87).

4.11 Creating F block types

S7 F Systems gives you the option of generating an F block type from the CFC chart of a safety program. It is possible to reuse F block types in other safety programs.

4.11.1 Rules for F-block types

The basic procedure for creating a new F block type is the same as in the standard user program. The same rules apply as for creating standard block types in CFC. Please also note:

- The new F block type can only contain F blocks from the F library, with the following exceptions:
 - F channel drivers for F signal modules
 - F blocks for F communication
 - F blocks F_CHG_BO, F_CHG_R, and F_MOV_R
 - F-control blocks
 - F-system blocks (of these, only the F_START block may be used)
- The F blocks called in the new F block type, as well as all F blocks in the safety program in which the F block type is being used, must come from the same library version. F blocks from different versions of the F library are not permitted.
- You are not permitted to connect one F block output to two chart I/Os.
- Within an F block type, the run sequence is not corrected automatically during compilation. The sequence defined on creation is retained.

Note

If the run sequence differs from the data flow (due to feedback, for example), compilation of the F block type will be aborted with an error. It is not possible for feedback to occur within an F block type.

- The chart I/Os of the new F block type can have both F data types and standard data types.
- You are not permitted to use the names of F blocks in the F library for the names of F block types.
- We recommend assigning names for instances of F blocks called in a single F block type as follows:
 - Numbers only as per the CFC editor default or alphanumeric names (but these must always start with "F_")
 - Capital letters only
 - No "_" at the end

Note

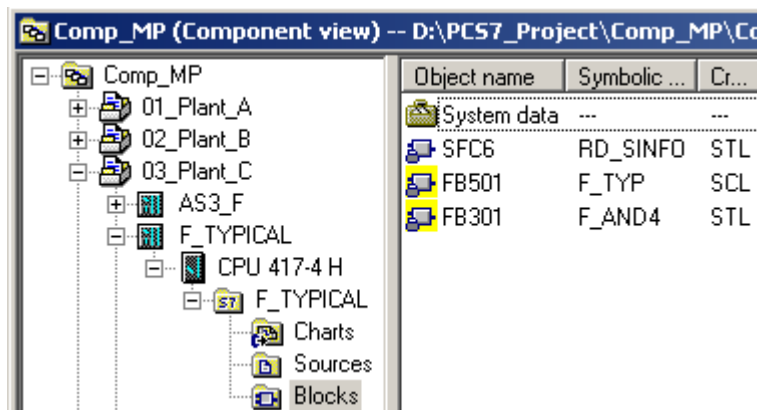
Outputs of F blocks always use the default initial values.

When creating F block types, you are not permitted to change initial values at outputs of F blocks. Although CFC does allow this and displays the change that has been made, S7 F Systems always uses the initial values explained in the description of the F blocks under "Default".

4.11.2 Creating F block types using "Compile chart as F block type"

Procedure

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



Note

As per the standard CFC procedure, when creating an F block type always use a separate AS station containing only the safety program of the F block type.

If you are using a CFC version earlier than V6.1, do not compile these charts as a program (menu command "Chart > Compile > Charts as Program"). Failure to do this can render the new F block type defective, as it erroneously contains data from the project in which it was created. This can cause errors in your safety program and even lead to an F-STOP of the safety program.

2. Under "Object Properties > Protection" in the CPU, select the "CPU contains safety program" function. The CPU (typical CPU) does not have to exist in the physical sense.
3. In this CPU, create a CFC chart and your function.

4. Select the menu command "Chart > Compile > Chart as Block Type". A dialog for entering the block properties appears.

The screenshot shows the 'Compile chart as block type' dialog box. The 'General' tab is selected. The 'Project path' is '03_Plant_CV\F_TYPICAL\CPU 417-4 H\F_TYPICAL\Blocks'. The 'Properties of Block Type' section includes: FB number: 501, Symbolic name: F_TYP, Name (header): F_TYP, Family: (empty), Author: (empty), Version (header): 1.0. The 'Compile for CPU' section has 'S7_400' selected. The 'Optimize Code for' section has 'Downloading changes in RUN' selected. The 'Know-how protected' checkbox is unchecked. Buttons for 'OK', 'Cancel', and 'Help' are at the bottom.

5. Enter the properties of the new F block type. Make sure that the names under "Symbolic name" and "Name (header)" are identical.
6. Select the "Compile for CPU - S7-400" and "Optimize Code for - Downloading changes in RUN" options and confirm by clicking OK.

Know-how protected is always checked regardless of the setting of the option.

Result: A new F block type which you can use in a safety program is created.

7. Add the new F block type to a safety program, along with the F block it calls, and test it there.
8. If function changes are required, change the block typical in the typical CPU and compile it again. Copy the block in order to test it again in your CPU.

Note

Do not perform any tests in the typical CPU: This CPU must only be used to generate block typicals from CFC charts. Do not perform compilation in the CFC editor.

Attributes whose names start with "F_" are managed by S7 F Systems.

You must assign different names to your own attributes to prevent deletion or overwriting during compilation.

Modifying F block types

You must update modified F block types in the CFC editor in the same way as all other block types. To do this, open the "Block Types" dialog with the menu command "Options > Block Types" and click "New version".

Making changes to F block types which have already been used may mean that you will need to subsequently compile and download the entire S7 program.

If you wish to use a new version of the F library, you must compile the F block types with this new version of the F library.

See also

<http://support.automation.siemens.com/WW/view/en/2201072>

4.12 Monitoring times and system response times

A fail-safe system must possess the ability to switch the process to a safe state within a defined reaction time in the event of a system error (e.g. CPU failure, F signal module failure, communications failure, program execution error).

To achieve this goal, a variety of monitoring times have been introduced to ensure that the affected functions are brought to a safe state if this time is exceeded.

What is the purpose of monitoring times and system reaction times?

For the S7-400 F/FH, there are essentially two different types of monitoring/error response:

- If the monitoring times of the fail-safe program are exceeded, the affected F program switches off and the inputs/outputs of the F signal modules are passivated.
- If the monitoring time for PROFIsafe communication between the F CPU and F signal modules is exceeded, the inputs/outputs of the affected F signal modules are passivated and the corresponding substitute values are forwarded to the CPU.

Both the availability and the safety of the F/FH system must be taken into account when configuring monitoring times.

- **Availability:**
The monitoring times must be set sufficiently high to prevent time monitoring from being triggered when no errors are present.
- **Safety:**
The monitoring times must be set sufficiently low to prevent the process safety time from being exceeded.

The rule of thumb is that a non-redundant S7 F/FH system supports shorter monitoring times and, therefore, shorter reaction times in the event of an error. For a redundant system, longer monitoring times have to be configured to allow for switchover of the components (PROFIBUS, interface modules) and the coupling/updating of the H-CPU.

Calculation

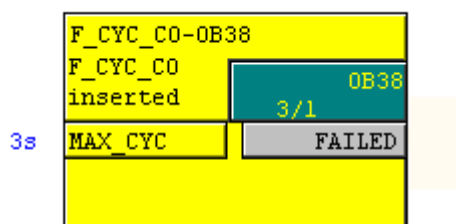
The Excel file "s7ftimea.xls" is used for the purpose of calculating the monitoring and reaction times below. You can find this file on the Internet by pointing your browser to the following link: (<http://support.automation.siemens.com/WW/view/en/22557362>)

The Excel file consists of the following sheets:

- max. runtime F-SG (Lib V1_3)
Here, you will find the runtimes of the F blocks from F library V1_3 in the various types of CPU. You can use these to estimate the runtime of your F program during the configuration stage of your project.
- max. runtime F-SG (Lib V1_2)
Here, you will find the runtimes of the F blocks from F library V1_2 in the various types of CPU. You can use these to estimate the runtime of your F program during the configuration stage of your project.
- min. F-specific monitoring times
This sheet contains formulae for calculating minimum values:
 - MAX_CYC
Maximum time between 2 calls of the cyclic interrupt OB with F program
 - PROFIsafe monitoring time
Maximum time between 2 frames from the master to the F I/O.
 - TIMEOUT between F-shutdown groups
Timeout value for fail-safe communication between F-shutdown groups in an F CPU
 - TIMEOUT between F CPUs
Timeout value for fail-safe communication between two F CPUs
- max. response times
You can use this to determine the maximum runtime of a signal from an input to an output in the system, with various types of configuration taken into account.

4.12.1 Calculating the F-cycle monitoring time (for block F_CYC_CO)

The F CPU runs execution time monitoring for every cyclic interrupt OB (OB 30 - OB 38) containing F-runtime groups.



Procedure

The first time the F program is compiled, you will be prompted to enter a value for the maximum cycle time (MAX_CYC) which may elapse between two calls to this cyclic interrupt OB. The default for the maximum F cycle time is 3,000 ms.

Should this default value not be suitable for your process, you can modify it directly at the F_CYC_CO block (located in the automatically generated @F_CycCo-OB3x chart).

You will need the following parameters for an exact calculation of the scan cycle monitoring time for a cyclic interrupt OB:

T _{CI}	Configured execution time of the cyclic interrupt OB in which the F_CYC_CO F function block is called.
T _{P15}	Configured "max. disabling time for priority classes > 15" Only relevant for redundant systems. Enter a 0 if you are not using an FH system.
Special handling	If the OB is entered as a cyclic interrupt OB with special handling, set "Yes" here.
T _{Prog}	Program runtime of the cyclic interrupt OB
T _{CiR}	CiR synchronization time
T _{CiRmax}	Upper limit of the CiR synchronization time (default setting: 1 s)
T _{CImax}	Shortest possible time for scan cycle monitoring of the cyclic interrupt OB

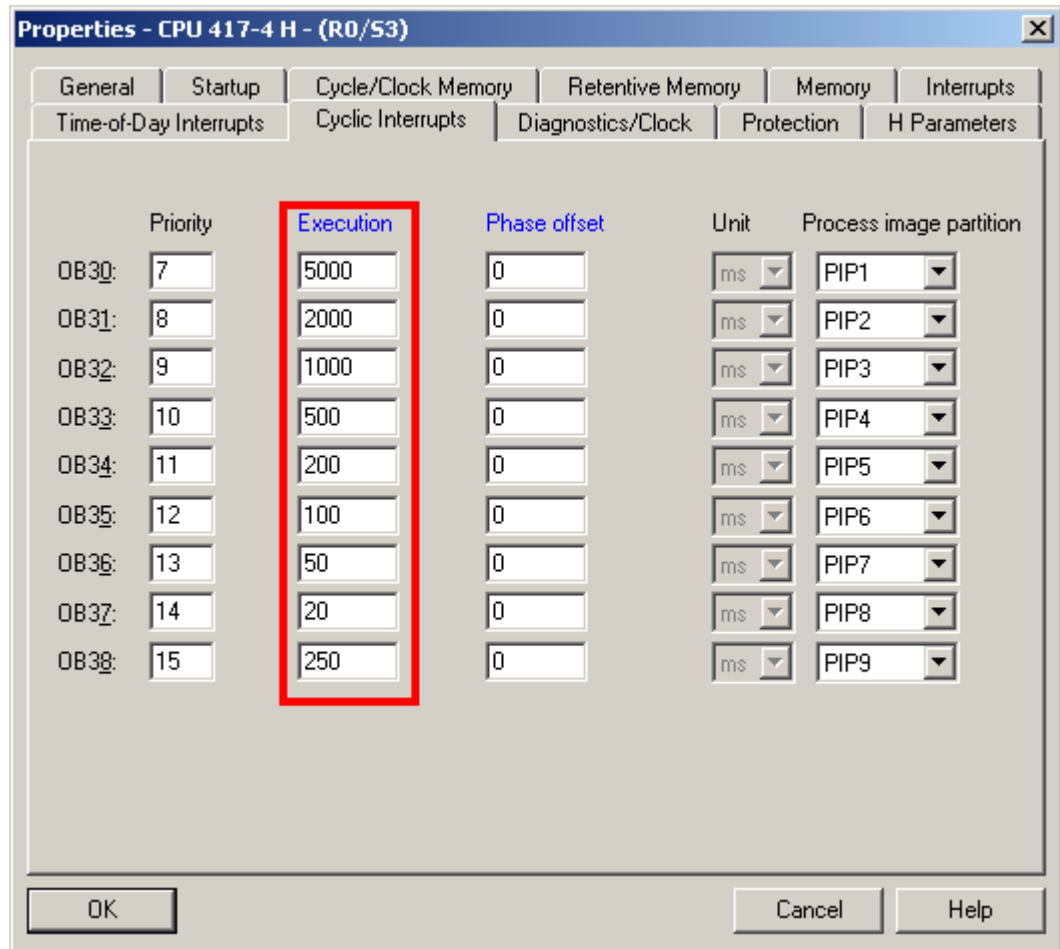
Extract from the "s7ftimea.xls" Excel file:

21	Configuring the monitoring time of the F-cycle time	
22	T _{CI}	300 ms
23	T _{P15}	800 ms
24	Special handling	Yes
25	Runtime of the program execution in the cyclic interrupt OB	30 ms
26	T _{CiR}	0 ms
27	T _{CiRmax}	0 ms
28	Minimum value for MAX_CYC	T_{CImax} 830 ms

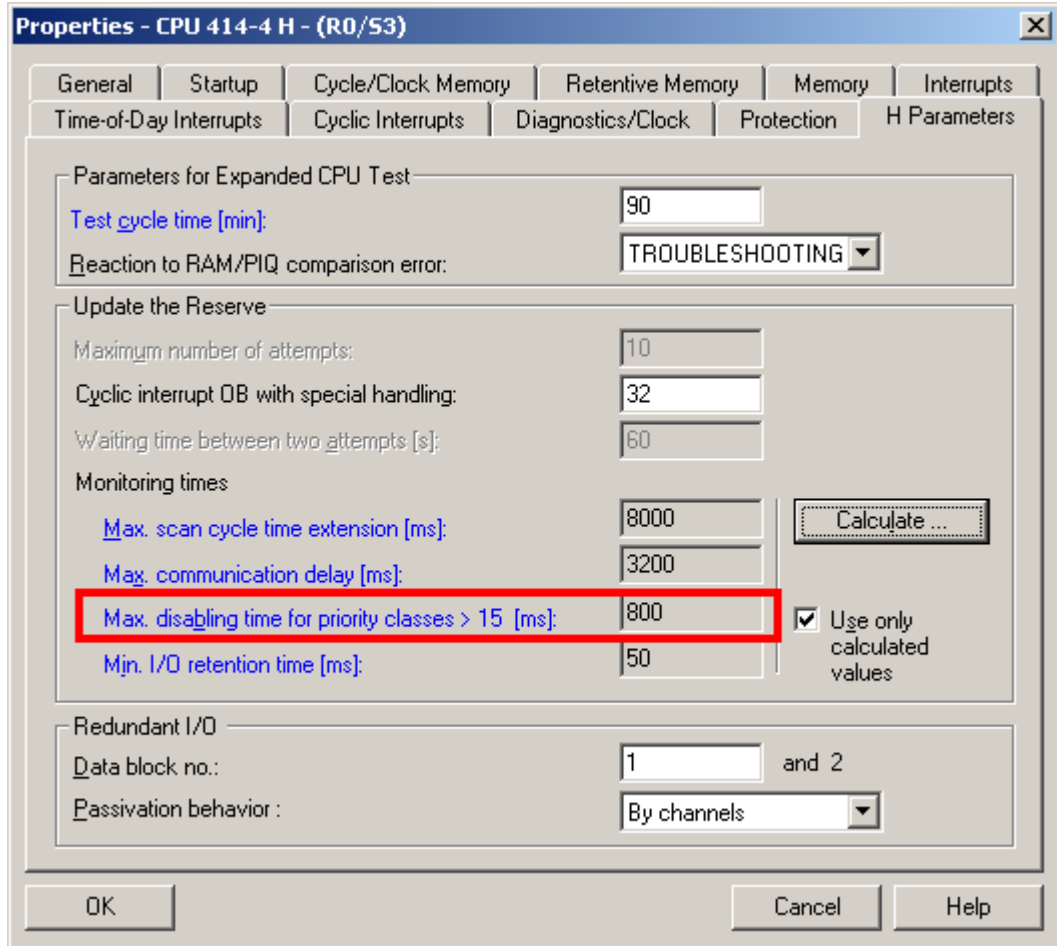
To prevent monitoring from being triggered when no errors are present, MAX_CYC must be set to a value which is higher than the maximum cycle time T_{CImax} of the corresponding cyclic interrupt OB.

- For non-redundant S7 F/FH systems:
 - The setting for T_{CImax} must be at least as high as the configured execution time (T_{CI}) of the cyclic interrupt OB.
- For redundant S7 F/FH systems:
 - In the S7 FH-system, the maximum disabling time for priority classes > 15 (T_{P15}) must also be taken into account in the context of updates.

You can find the value for T_{CI} in the "Execution" column of the relevant cyclic interrupt OB, located on the "Cyclic Interrupts" tab in the CPU properties.



Parameter T_{P15} is the same as the "Max. disabling time for priority classes > 15" parameter from the H parameters of the CPU.

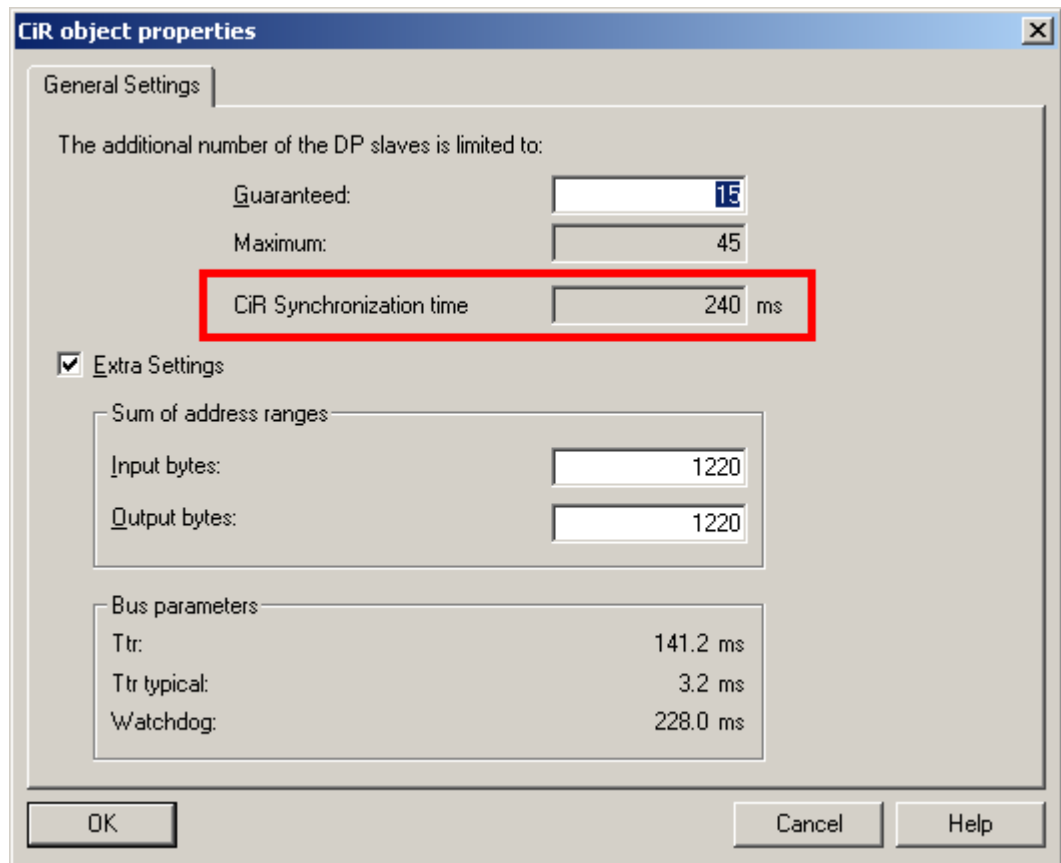


If the cyclic interrupt OB in which the F_CYC_CO block is called has been entered as a "cyclic interrupt OB with special handling", set "Yes" for special handling in the Excel table.

You only have to enter parameters T_{CiR} and T_{CiRmax} if you are not using an H system and have activated the CiR function. If you are not using CiR, enter 0 for both values.

T_{CiRmax} is set to 1 s in the system, but can be changed by calling SFC 104.

Parameter T_{CiR} can be found in the singular system with the CiR function activated, in the "Properties" dialog of the CiR object created.



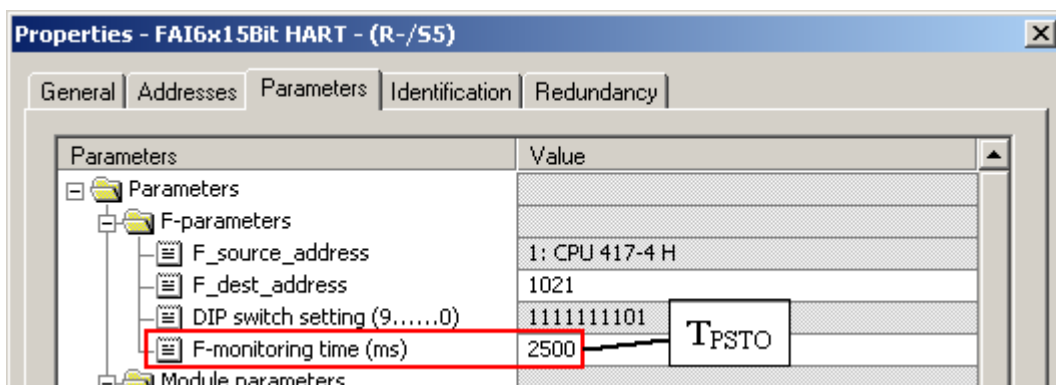
Note

For more information on CiR, refer to the "Modifying the System during Operation via CiR" Manual. (<http://support.automation.siemens.com/WW/view/en/14044916>)

4.12.2 Communication monitoring time for F CPU/F signal modules

Time monitoring of PROFIsafe communication is implemented in the F signal modules and in the F CPU using F module drivers.

The value is entered while assigning parameters for the F signal modules in HW Config ("F_monitoring time" parameter) and applied automatically when the F module drivers are generated.



To prevent either monitoring in the F module driver or monitoring in the F signal module from being triggered when no errors are present, the F monitoring time T_{PSTO} must be set to a sufficiently high value.

You can determine the shortest possible F monitoring time using S7ftimea.xls.

31	Configuring the PROFIsafe monitoring time		
32		Variants 1 to 6:	2
48	minimum PROFIsafe monitoring time	T_{PSTO}	1067 ms

Depending on the hardware configuration in question, S7ftimea.xls distinguishes between 6 different options:

- Variant 1:
Singular system with F signal modules in a PROFIBUS DP slave
- Variant 2:
Redundant system with F signal modules in a switched (ET 200M) PROFIBUS DP slave
- Variant 3:
Singular system with F-PA devices on a PROFIBUS DP/PA coupler
- Variant 4:
Singular system with F-PA devices on a PROFIBUS DP/PA link
- Variant 5:
Redundant system with F-PA devices on a switched PROFIBUS DP/PA link
- Variant 6:
Redundant system with F-DP devices on a switched PROFIBUS Y link

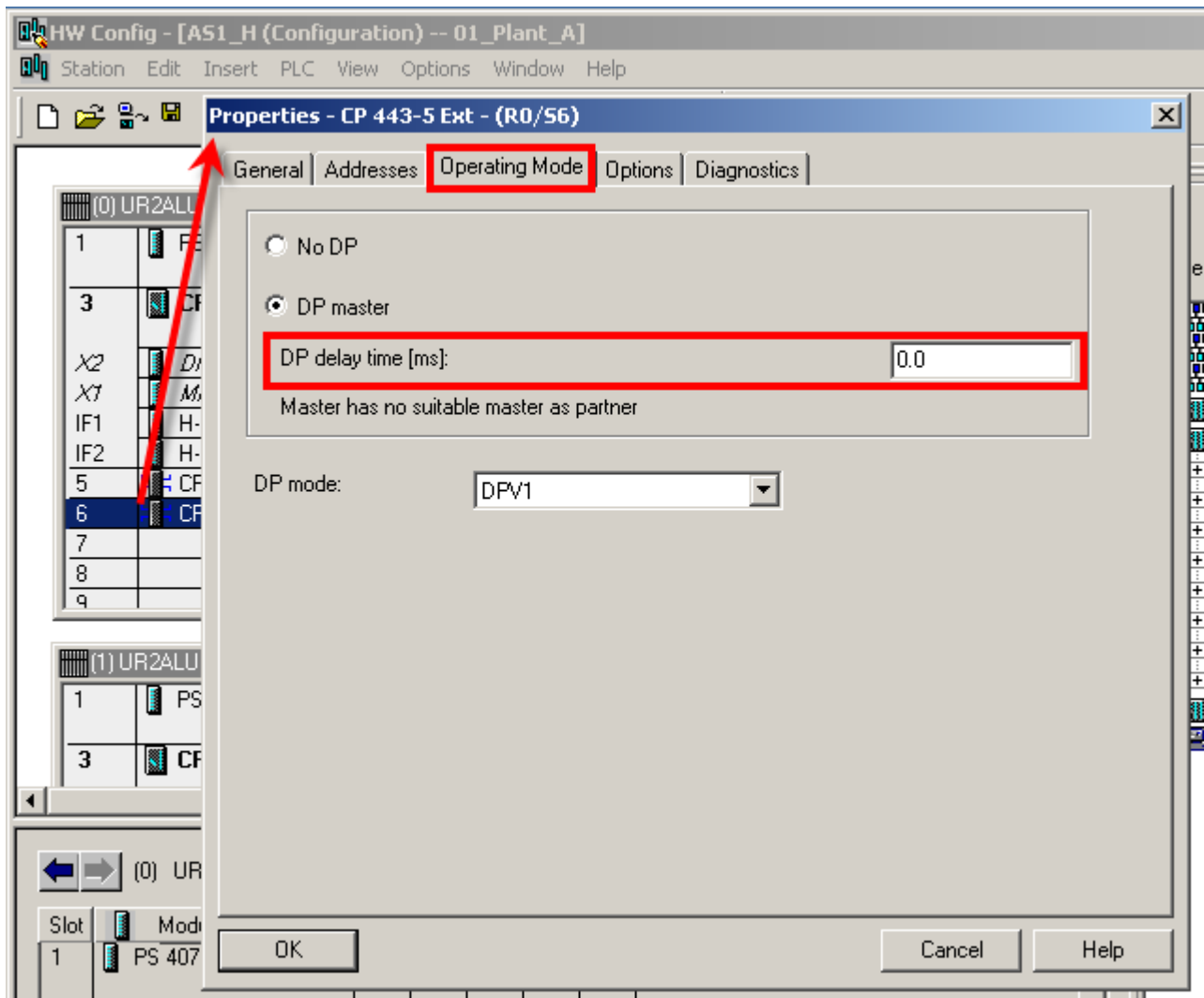
The F monitoring time must be calculated for the relevant variant of each F signal module type or device, and the values for the cyclic interrupt OBs must be calculated as well. To do this, select the appropriate configuration variant and enter the parameters in the corresponding line.

Variant 2: single-channel switched I/O or redundant switched F-I/O via PROFIBUS-DP	CPU (FH-System)			CP	PROFIBUS-DP	IM	F I/O	F-I/O with inputs and outputs?
	T_{CI}	T_{Cmax}	T_{FPROG}	T_{DP_DLY}	T_{TR}	T_{slave}	T_{DAT}	
	300 ms	830 ms	30 ms	0 ms	18 ms	1 ms	29 ms	No
	TFPROG is not taken into account			T_{DP_FD}	T_{DP_SO}	T_{SLAVE_SO}		
				170 ms	140 ms	30 ms		

You will need the following parameters for the calculation:

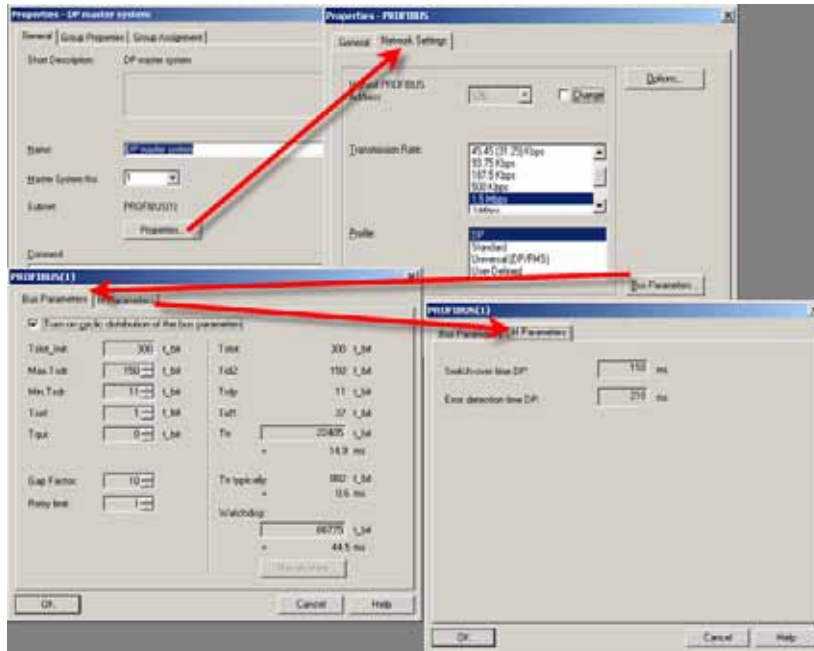
T_{CI}	Configured execution time of the cyclic interrupt OB in which the F module driver is processed
T_{PROG}	Program runtime of the cyclic interrupt OB in which the F module driver is processed
T_{Cmax}	Time set at the F_CYC_CO block for scan cycle monitoring of the cyclic interrupt OB
T_{DP_DLY}	Additional delay time due to external DP interface (via CP 443-5 Extended)
$T_{TR/TR1}$	Maximum target rotation time for the DP master system
T_{TR2}	Maximum target rotation time for the subordinate DP master system on the Y link, or maximum target rotation time for the subordinate PA master system on the DP/PA coupler
$T_{DPPA_L_DLY}$	Delay time due to a DP/PA link
$T_{Y_L_DLY}$	Delay time due to a Y link
T_{slave}	Maximum delay time due to IM 153-2 (typical value: 1 ms)
T_{DAT}	Maximum acknowledgment time of the fail-safe I/O module in safety mode (F signal module group data sheet)
With inputs and outputs	If you are using F signal modules with inputs and outputs, set "Yes" here (e.g. ET 200pro). If you are using an ET 200M, set "No".
T_{DP_FD}	Maximum DP error detection time Only relevant for redundant IM 153-2 (switched I/O). Enter 0 if you are not using a redundant IM.
T_{DP_SO}	Maximum DP switchover time Only relevant for redundant IM 153-2 (switched I/O). Enter 0 if you are not using a redundant IM.
T_{SLAVE_SO}	Maximum DP switchover time Only relevant for redundant IM 153-2 (switched I/O). Enter 0 if you are not using a redundant IM.
$T_{F-SIGNAL\ MODULE,\ ACK}$	Maximum acknowledgment time of the fail-safe I/O module in safety mode (F signal module group data sheet)
T_{DP_DLY}	Additional delay time due to external DP interface (via CP 443-5 Extended)

You can find the T_{DP_DLY} parameter on the "Operating Mode" tab of the CP 443-5 ext properties in HW Config.



$T_{TR1/TR2}/T_{DP_FD}/T_{DP_SO}$ parameter

You can find the parameters in the properties of the PROFIBUS DP or PA master system by double-clicking the bus line. In HW Config, the following screens from which the relevant values can be taken are shown:



T_{SLAVE_SO} parameter

T_{SLAVE_SO} depends on the DP master system and the IM being used.

DP master system	ET 200M with IM 153	Switchover time	Configuration
T_{SLAVE_SO}	-2AA02	70 ms	Any configuration
	-2AB01	30 ms	Without F, FM, or HART modules
	-2Bx00	30 ms	Any configuration
	-2Bxx1	30 ms	Without F, FM, or HART modules

$T_{DPPA_L_DLY}$ parameter

- Additional delay time due to a DP/PA coupler in a singular system
- Or additional delay time and switchover time due to a DP/PA link in the redundant system

Precondition	Switchover time
Switchover time with unchanged PA configuration	Typ.: 70 ms + number of PA field devices x 51 ms
	Max.: 820 ms + number of PA field devices x 50 ms
Switchover time when configuration is changed during operation	Typ.: 80 ms + number of PA field devices* x 67 ms
	Max.: 800 ms + number of PA field devices* x 130 ms

* With unchanged PA field device addresses

T_{Y_L_DLY} parameter

- Additional delay time due to a Y link in the redundant system

Precondition	Switchover time
Delay time due to a Y link	5 ms

T_{DAT} parameter

You will find the maximum acknowledgment time of the F signal modules in the corresponding data sheet for the I/O module concerned.

Module	Acknowledgment time in safety mode
SM326; DI x 24 DC 24V	With 1oo1 sensor evaluation: max. 29 ms
	With 1oo2 sensor evaluation: max. 30 ms
SM326; DI 8 x NAMUR	Max. 68 ms
SM326; DO 10 x DC 24V/2A	Max. 20 ms
SM326; DO 8 x DC 24V/2A PM	Max. 14 ms
SM336; AI 6 x 13Bit	Acknowledgment time = Maximum reaction time = Maximum reaction time per channel x N + maximum basic reaction time (N = Number of activated channels) <ul style="list-style-type: none"> • Response time per activated channel: <ul style="list-style-type: none"> – At 50 Hz: max. 50 ms – At 60 Hz: max. 44 ms • Basic response time: <ul style="list-style-type: none"> – At 50 Hz: Max. 50 ms – At 60 Hz: Max. 44 ms
SM336; F-AI 6 x 0/4 ... 20 mA HART	Acknowledgment time = Conversion cycle time = (Basic reaction time + N x reaction time per channel pair) x filter (N = Number of activated channel pairs)
	<ul style="list-style-type: none"> • Reaction time per activated channel pair: <ul style="list-style-type: none"> – At 50 Hz: Max. 30 ms – At 60 Hz: Max. 27 ms • Basic reaction time: 23 ms • Filter for measured values (per channel): <ul style="list-style-type: none"> – None: 1 x cycle time – Weak: 4 x cycle time – Medium: 16 x cycle time – Strong: 64 x cycle time

4.12.3 Monitoring time for safety-related communication between F-CPU's

Introduction

Time monitoring of fail-safe communication between 2 F CPUs is implemented in the send and receive blocks F_SENDR and F_RCVR or F_SENDBO and F_RCVBO with the same TIMEOUT monitoring time, which needs to be configured on both the send and receive blocks. The lowest value for the timeout time can be determined using S7ftimea.xls.

95	Configuration of the TIMEOUT input of the F_SENDBO/F_RCVBO, F_SENDR/F_RCVR, or F_SDS_BO/F_RDS_BO F-FBs								
107	minimum value for TIMEOUT	3200 ms							
108									
109		CPU1				CPU2			
110	Communication	T _{CI1}	T _{Delay1}	T _{CIr1}	T _{CIrmax1}	T _{CIrmax2}	T _{CIr2}	T _{Delay2}	T _{CI2}
111	via S7 interconnections	300	2000	0	0	0 ms	0 ms	2000 ms	300 ms
112									

You will need the following parameters to calculate the "TIMEOUT" parameter:

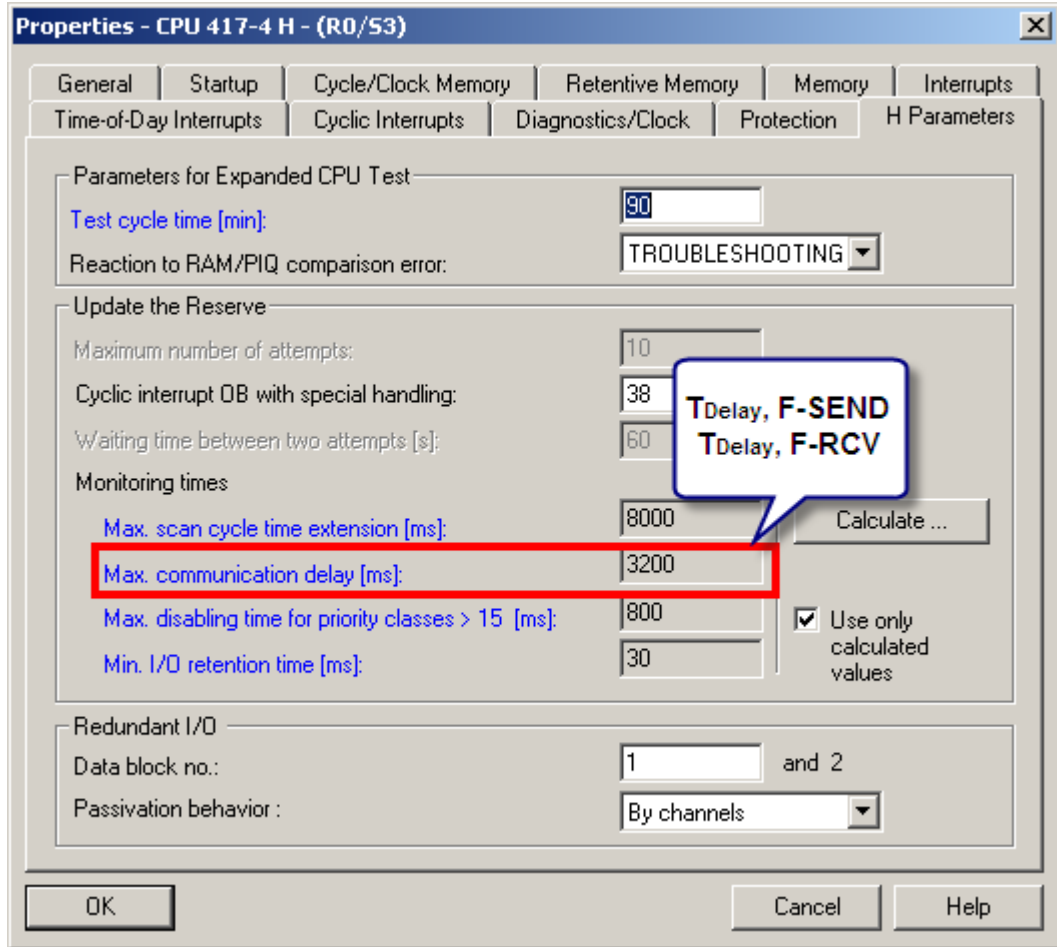
Communication between F CPUs (TIMEOUT parameter)	
CPU 1, sender	
T _{CI1}	Configured cycle time of the cyclic interrupt OB in which the send blocks F_SENDBO or F_SENDR are called.
T _{Delay1, F_SEND}	Maximum communication delay when updating the standby in the S7 FH system with call of F_SENDBO or F_SENDR. Only relevant for redundant S7 FH systems. Enter a 0 if you are not using an S7 FH system.
T _{CIr1}	CiR synchronization time
T _{CIrmax1}	Upper limit of the CiR synchronization time (default setting: 1 s)

CPU 2, recipient	
T _{CIrmax2}	Maximum communication delay when updating the standby in the FH system with call of F_RCVBO or F_RCVR. Only relevant for redundant S7 FH systems. Enter a 0 if you are not using an S7 FH system.
T _{CI2}	Configured cycle time of the cyclic interrupt OB in which the receive blocks F_RCVBO or F_RCVR are called.

With the exception of the T_{Delay1/Delay2} parameters, all the parameters are described in Section Monitoring times and system response times (Page 110).

T_{Delay1/Delay2} parameters

You can find these parameters in the H parameters within the CPU properties.

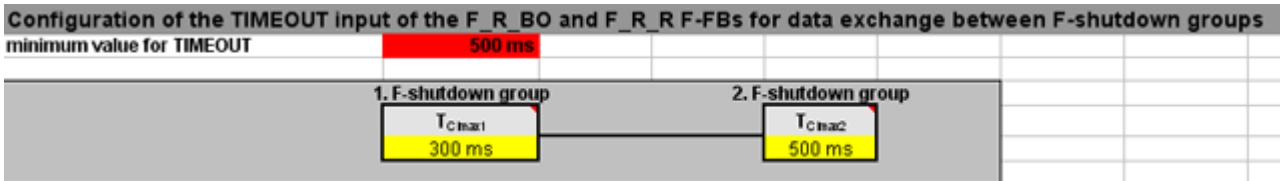


The value should be taken from the transmitting or receiving CPU.

4.12.4 Monitoring communication between F-shutdown groups

Time monitoring is implemented in the F_R_BO or F_R_R F FBs and configured at the "TIMEOUT" input parameter of the receive block.

To prevent time monitoring from being triggered when no errors are present, the TIMEOUT monitoring time must be set to a value which is at least equal to the higher of the two maximum cyclic interrupt cycle times of F_S_R or F_S_BO and F_R_R or F_R_BO. The lowest value for the timeout time can be determined using S7ftimea.xls.



You will need the following parameters to calculate the "TIMEOUT" parameter:

T _{C1max1}	Maximum cycle time of the cyclic interrupt OB in which the associated send block F_S_BO or F_S_R is called.
T _{C1max2}	Maximum cycle time of the cyclic interrupt OB in which the receive block F_R_BO or F_R_R is called.

The values for T_{C1max1}/T_{C1max2} can be found in chart @F_CycCo-OBxx, at input MAX_CYC of the F_CYC_CO-OBxx F function block (xx = Number of the cyclic interrupt OB with the send/receive block).

4.12.5 Response times of safety functions

Definition of response time

The response time is the time between the detection of an input signal and the changing of a linked output signal.

The actual response time is always between a minimum and a maximum response time. When configuring your plant, you must always assume the maximum response time.

The maximum reaction time of a safety function must be shorter than the process safety time.

Definition of process safety time

The process safety time is the time within which the process can be left to its own devices without creating a dangerous situation.

Within the process safety time, the S7 F/FH system controlling the process is not under control; in other words, it might malfunction or fail completely. The process safety time depends on the type of process and must be specified individually.

How to calculate the response time

In S7ftimea.xls, the route of a signal from the sensor to the actuator via the system can be broken down into five possible sections:

1. Input:
Runtime from sensor to user program
2. Processing in the 1st F CPU:
Cycle time and runtime of F program
3. F CPU-to-F CPU communication:
Sends the signal to a second F CPU (optional)
4. Processing in the 2nd F CPU:
Cycle time and runtime of F program in the second F CPU (optional)
5. Output:
Runtime from user program to actuator

Different variants may be used in each section. To calculate the maximum reaction time, select the relevant variant from the header line of each section and enter the corresponding values in the variants.

Input/Output

Depending on the hardware configuration of the system, 4 variants are possible here. In terms of the input and output, the variants are exactly the same, with the sole exception that the blocks for the output are in the reverse sequence to those for the input.

You will need the following parameters for calculation:

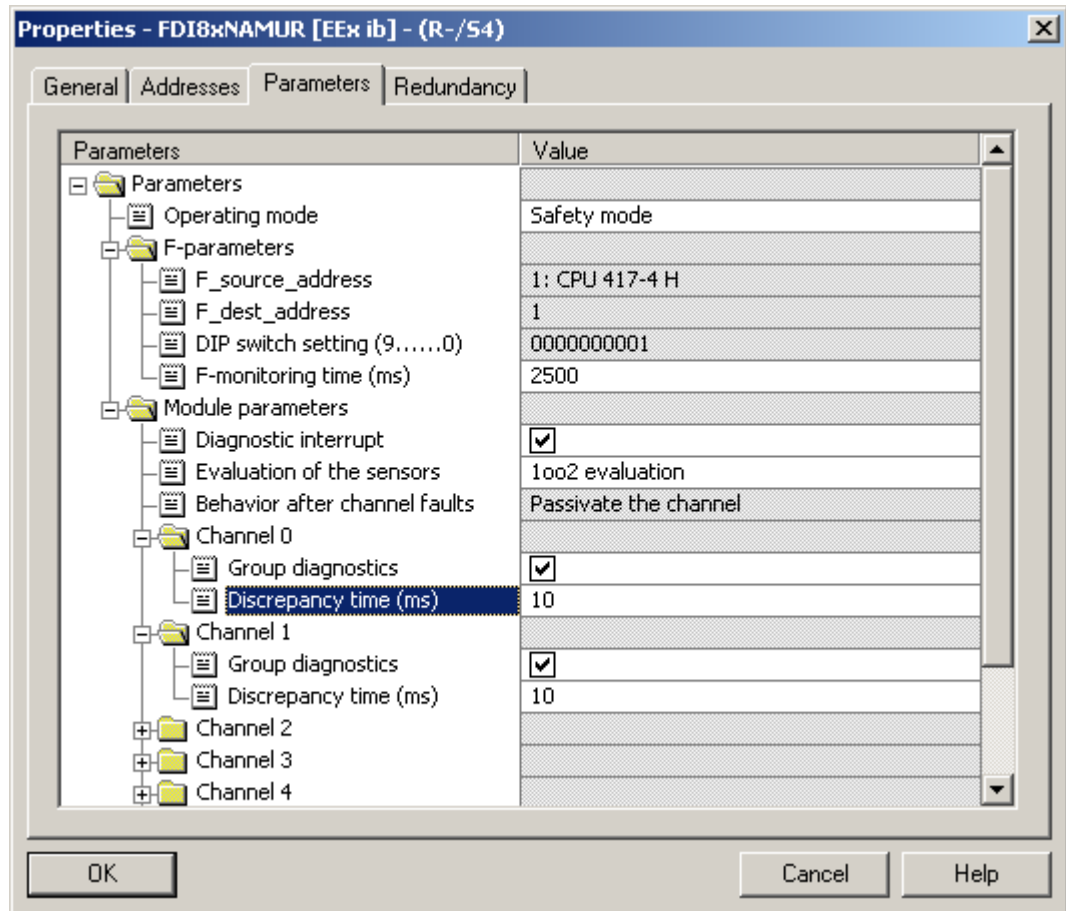
T _{Sensor_DLY} /T _{Actuator_DLY}	Delay time of sensor or actuator
T _{DIS}	Maximum discrepancy time
T _{WCDT}	Worst Case Delay Time Maximum reaction time when no errors are present
T _{OFDT}	One Fault Delay Time Maximum reaction time when an error is present
T _{DAT}	Maximum acknowledgment time of the fail-safe I/O module in safety mode (F signal module group data sheet)
T _{PSTO, configured}	Configured PROFIsafe monitoring time
T _{Slave}	Maximum delay time due to IM 153-2 (typical value: 1 ms)
T _{TR/TR1}	Maximum target rotation time for the DP master system
T _{TR2}	Maximum target rotation time for the subordinate DP master system on the Y link, or maximum target rotation time for the subordinate PA master system on the DP/PA coupler
T _{DPPA_L_DLY}	Delay time due to a DP/PA link
T _{Y_L_DLY}	Delay time due to a Y link
T _{DP_DLY}	Additional delay time due to external DP interface (via CP 443-5 Extended)

$T_{\text{Sensor_DLY}}/T_{\text{Actuator_DLY}}$ parameters

The delay times in the table are assumed to be 100 ms in each case; if necessary, the values for these will need to be obtained from the device data sheets.

 T_{DIS} parameter

You set the discrepancy time when 1oo2 (2v2) has been selected in the hardware configuration of the F signal module.

 **$T_{\text{WCDT}}/T_{\text{OFDT}}$ parameters****Note**

Whether errors are present or not, the maximum reaction times depend on the type of module, and can be obtained from the "Automation System S7-300 ET 200M Distributed I/O Device Fail-safe Signal Modules" Manual:

<http://support.automation.siemens.com/WW/view/en/19026151>

4.12 Monitoring times and system response times

Enter 0 if:

- No 1oo2 (2v2) evaluation is performed on the module.
- The value 0 is immediately output in the event of a discrepancy.

F signal module	T _{WCDT}	T _{OFDT}
6ES7326-1BK01-0AB0 DI 24 x DC 24 V	= Internal processing time + input delay + short-circuit test time = 23 ms + 3 ms + 6 ms = 32 ms	= 31 ms with 1oo1 evaluation = 29 ms with 1oo2 evaluation
6ES7326-1RF00-0AB0 DI 8 x NAMUR	= Internal processing time + input delay = 60 ms + 3 ms = 63 ms	
6ES7326-2BF40-0AB0 DO 10 x DC 24V/2A	= Internal processing time + input delay = 24 ms + 0 ms = 24 ms	
6ES7326-2BF01-0AB0 DO 8 x DC 24V/2A PM	= Internal processing time = 10 ms	
6ES7336-1HE00-0AB0 AI 6 x 13Bit	= N × reaction time per activated channel + basic reaction time <ul style="list-style-type: none"> • E.g.: N = 6, interference frequency of 50 Hz = 6 × 50 ms + 50 ms = 350 ms • E.g.: N = 6, interference frequency of 60 Hz = 6 × 44 ms + 44 ms = 308 ms 	
6ES7336-4GE00-0AB0 F-AI 6 x 0/4 ... 20 mA HART	With 1oo1 (1v1): = 2 × conversion cycle time × filter <ul style="list-style-type: none"> • E.g.: interference frequency of 50 Hz, filter = 1 conversion cycle, 3 active channel pairs = 2 × (3 × 25 ms + 50 ms) × 1 = 250 ms 	= 2 × conversion cycle time <ul style="list-style-type: none"> • E.g.: interference frequency of 50 Hz, 3 active channel pairs = 2 × (3 × 25 ms + 50 ms) = 250 ms
	With 1oo2 (2v2): = 2 × conversion cycle time × filter + 2 × conversion cycle time <ul style="list-style-type: none"> • E.g.: interference frequency of 50 Hz, filter = 1 conversion cycle, 3 active channel pairs = 2 × (3 × 25 ms + 50 ms) × 1 + 2 × (3 × 25 ms + 50 ms) = 500 ms 	

The discrepancy time is not contained in T_{OFDT}. It should always be entered at parameter T_{DIS}.

Other parameters

The remaining parameters have been discussed in the preceding sections of this chapter.

Processing in the 1st CPU/Processing in the 2nd CPU

Processing in the 2nd CPU is optional. It involves the same variants as in the 1st CPU.

There are 2 different variants:

- One F-shutdown group is present in the signal flow being examined.
- Two F-shutdown groups are present in the signal flow being examined.

You will need the following parameters for calculation:

$T_{CI}/T_{CH1}/T_{CI2}$	Configured cycle time of the cyclic interrupt OB in which the F function block has been installed in each case
$T_{CI\max, \text{proj.}}/T_{CI\max1, \text{proj.}}/T_{CI\max2, \text{proj.}}$	Time set in chart @F_CycCo-OBxx, at input MAX_CYC of the F_CYC_CO-OBxx F function block (xx = Number of corresponding cyclic interrupt OB).
$T_{FPROG}/T_{FPROG1}/T_{FPROG2}$	Program runtime of the corresponding cyclic interrupt OB
TIMEOUT	Time parameterized at the TIMEOUT input of the F_R_BO or F_R_R block

CPU-to-CPU communication (optional)

If your signal is transferred from one CPU to another, select "Yes" in the header. For calculation purposes, you require the time parameterized at the TIMEOUT input of the send and receive blocks for communication.

If CPU-to-CPU communication is not taking place, leave the default setting ("No") unchanged. The next section, "Processing in the 2nd CPU", will not be taken into account during calculation in this case.

Result

As the result, S7ftimea.xls returns two groups with 3 times each.

max. response time from the input terminal of the F-I/O (input) to the output terminal of the F-I/O (output)			
if there are no faults/errors		461 ms	
if there is a fault/error		1748 ms	
for any run time of the standard or fault-tolerant system		3571 ms	
max. response time from sensor to actuator			
if there are no faults/errors		661 ms	
if there is a fault/error		1948 ms	
for any run time of the standard or fault-tolerant system		3771 ms	

The lower group takes into account the delay times of the sensor and actuator.

Check that the times calculated meet the requirements of your process.

Configuration with Safety Matrix

5.1 Creating and configuring a Safety Matrix

Note

You can find information on creating and configuring a Safety Matrix in the "*Industrial Software Safety Matrix*" Manual.

(<http://support.automation.siemens.com/WW/view/en/19056619>)

A Safety Matrix is determined by its version number, file version, and signature. Each time the matrix is saved, the file version changes along with (depending on the change made) the minor version number and signature. Both values are downloaded to the AS. During online access from the engineering tool or OS, the values are compared and a message is output if any differences are detected. If a matrix with a modified version number and signature is downloaded to the CPU, the OS server and single stations will need to be compiled and downloaded as well.

5.2 Documenting a Safety Matrix

A Safety Matrix is documented via menu item "Options > Reports > Configuration Report".

This displays all of the configurations in the matrix in a text file; they can then be printed or saved in this format. The same matrix can be reconfigured on the basis of this text file,

or a hard copy of it can be printed out via the menu item "File > Print...". In order to ensure that as much information as possible is retained in this printout, activate all options on the "General" tab prior to printing (accessible via "View > Adapt > Layout"); otherwise, fields that are not visible will not be printed. Please note that this type of printout does not meet the criteria of an acceptance document, as it lacks key information such as a signature and cyclic interrupt OB.

5.3 Organizing matrices into different shutdown groups

The information below is intended to demonstrate how two matrices located in different shutdown groups but the same OB are installed. This is carried out using S7 F Systems Lib V1_3.

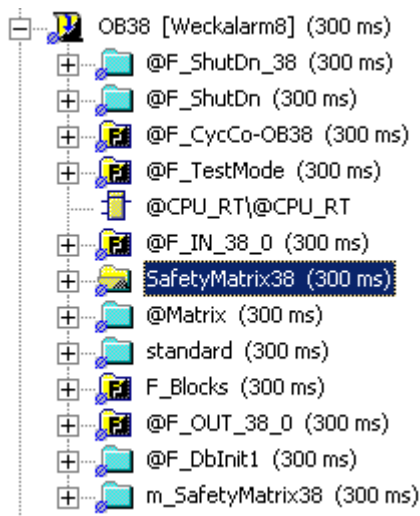
Creating shutdown groups

The F module driver of an F signal module may only be assigned to one shutdown group.

Therefore, when creating shutdown groups you must make sure that all signals in an F signal module are used in the same shutdown group. Only one module driver is generated per F signal module, which, if signals in an F signal module are used in different shutdown groups, can lead to compilation errors on account of invalid interconnections.

In the example below, the intention is to have two matrices be executed with a matrix cycle time of 300 ms, but in separate shutdown groups.

The starting point is the existing program, including the first matrix to have been compiled.

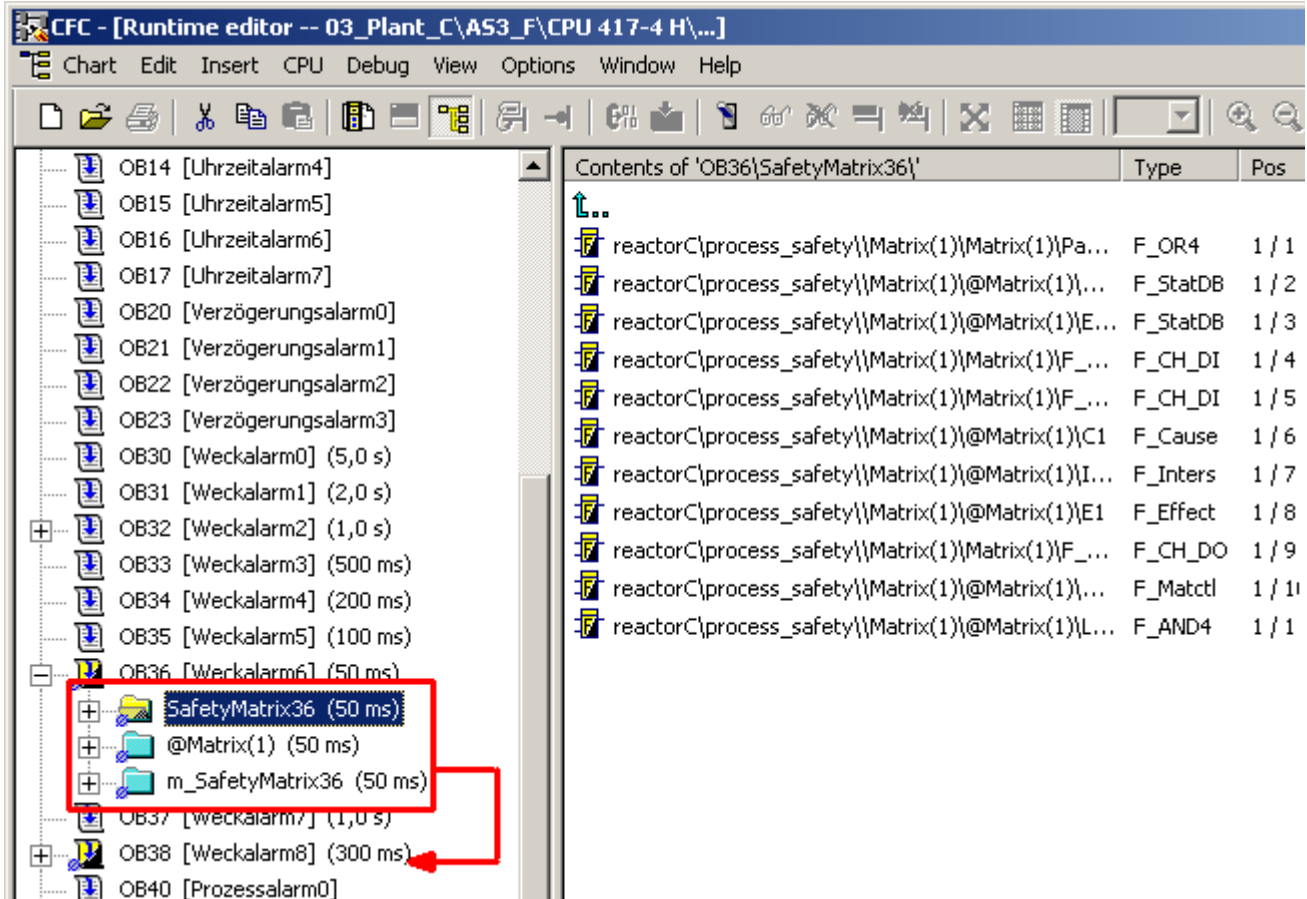


Inserting a new matrix in a new shutdown group.

To insert a new matrix and assign it to a new shutdown group, proceed as outlined below.

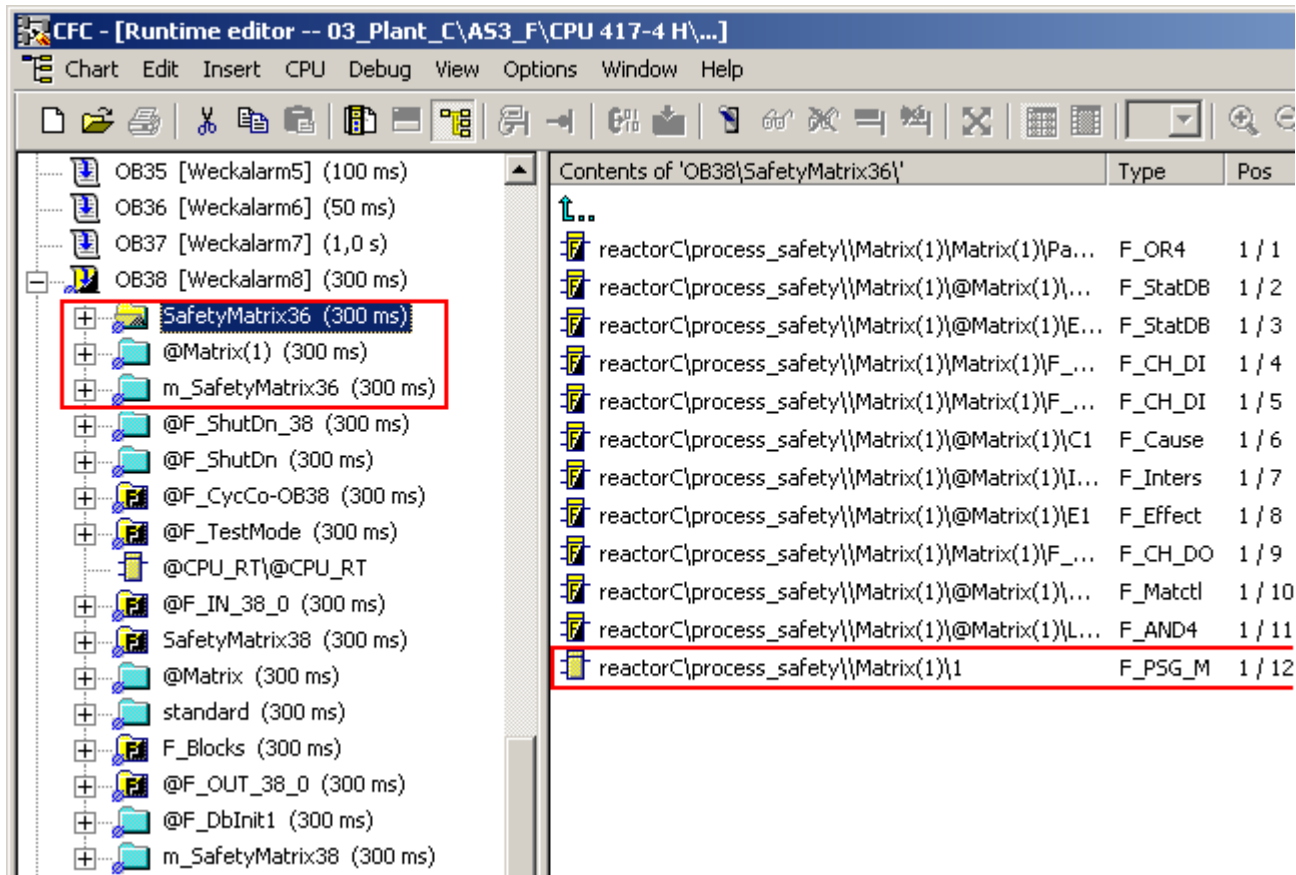
1. Create a new matrix and, in the Safety Matrix engineering tool, for the matrix cycle time select a cyclic interrupt OB that is not in use and into which no matrices have yet been inserted.
2. Create your safety function and transfer the matrix. In the cyclic interrupt OB, three runtime groups will be created for this matrix.

3. Move all three runtime groups to the destination cyclic interrupt OB (in this case, cyclic interrupt OB 38), placing them at the starting point of the OB.



5.3 Organizing matrices into different shutdown groups

- Place the F_PSG_M block in the matrix chart to create the required shutdown group. Make sure to insert it at the end of the new matrix runtime group.



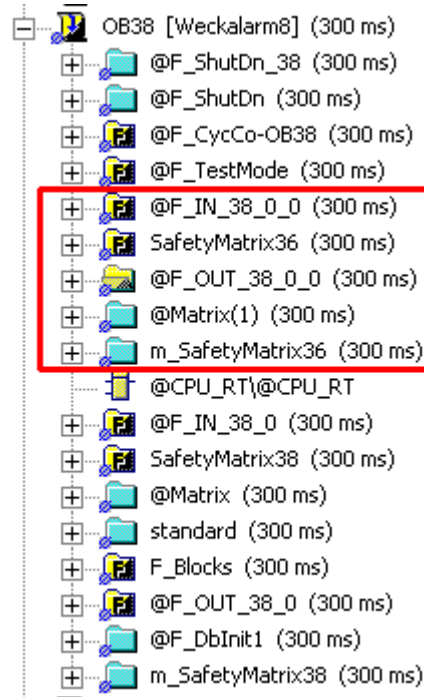
- In order to transfer any changes that may be made in the Safety Matrix engineering tool to the correct runtime/shutdown group as well, you will then need to set the matrix cycle time to the current value, which in this case is 300 ms (OB 38).

- Transfer the matrix with the modified cyclic interrupt OB setting to the project. When changes are transferred, the engineering tool will locate the runtime group that has been created and moved, and transfer changes to it as well. The following information appears briefly in the status bar of the Safety Matrix engineering tool:

The Matrix Runtime Group has been renamed (SafetyMatrix35) [This may be changed during transfer]

- Compile the S7 program.

The compiler will create the newly generated shutdown group and the additional runtime groups @F_IN_<OBNr>_xx and @F_OUT_<OBNr>_xx.



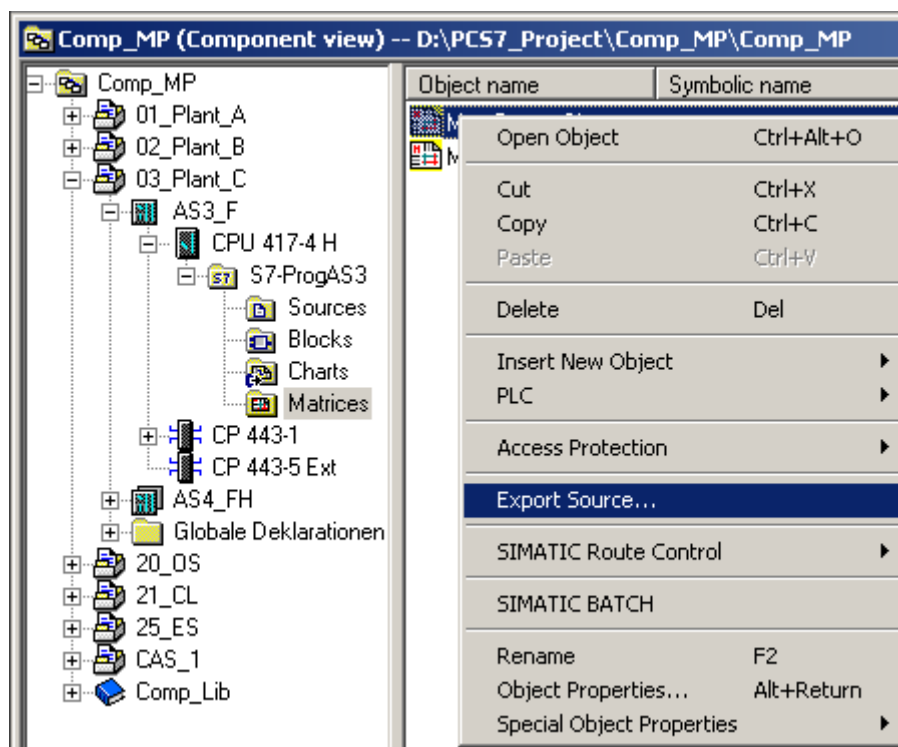
Note

Please note that if the matrix CFC shutdown groups are subsequently moved or the safety program is subsequently divided into shutdown groups, the amount of additional effort may be considerable in terms of the change configuration work involved. For this reason, you must make sure that division into partial shutdown groups takes place at the start of configuration and that the F signal modules are assigned to shutdown groups.

5.4 Duplicating matrices

In order to keep the same selected settings, such as SIF assignments (safety instrumented function groups), in another matrix, it is possible to duplicate the matrix template. To do this, proceed as follows:

1. In the matrix folder of the CPU, select the matrix you wish to use as a template and export it as a source via the context menu.



2. Select a destination folder and save the file here under a new name. Renaming the file is important as, if you do not do this, your template file will be overwritten when you carry out an import.
3. To import a template file of this kind, open the Object Properties of the matrix folder. You can use the "Import CEM" button to add the saved matrix template to a matrix folder.
4. In the dialog shown below, enter a new name for the matrix; this must be different from the one assigned to the template.

5.5 User authorizations for the Safety Matrix Viewer

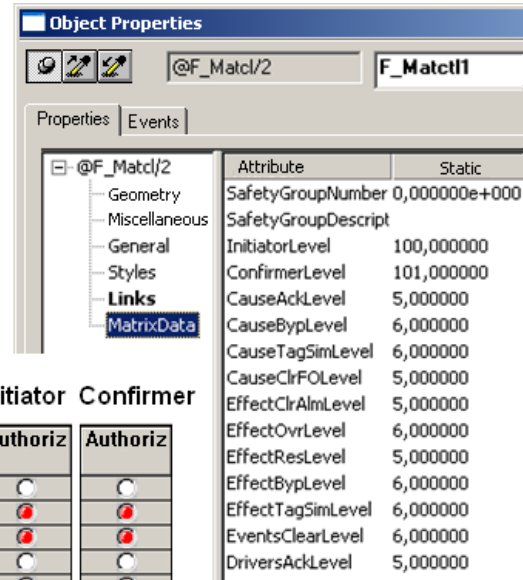
In the properties of the Safety Matrix block icon, you can set up a separate user authorization for each action and assign it to selected users.

Here, it is also possible to enter the authorization level for the initiator and confirmer for each OS operation. To split operator authorizations across two different users (i.e. for doubly-assured authorization by two users), you can assign the initiator level to one user and the confirmer level to the other,

meaning that each relevant matrix transaction must be initiated by the first user and confirmed by the second.

The example configuration below involves a "user" who has process operator authorization (authorization level 5) and therefore possesses both "initiator" (authorization level 100) and "confirmer" (authorization level 101) rights at this level.

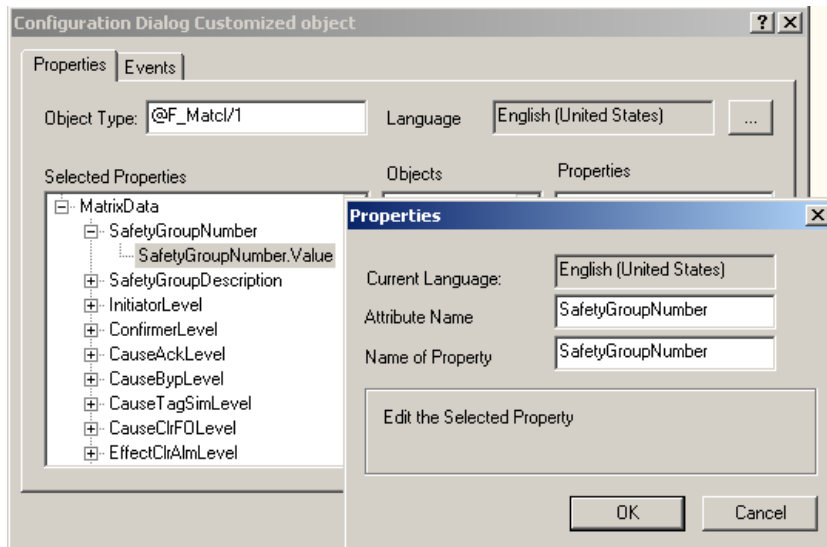
However, this "user" is only permitted to execute acknowledgments. In order to execute "higher-level process operations" (authorization level 6) in the matrix, an "initiator" (authorization level 100) and "confirmer" (authorization level 101) are required.



No.	Function	Authoriz	Initiator	Confirmer
1	User administration	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
2	Authorization for area	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>
3	System change	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>
4	Monitoring	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
5	Process controlling	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>
6	Higher process controlling	<input type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>
7	Report system	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
100	Initiator	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
101	Confirmer	<input checked="" type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

The values set in the matrix faceplate under MatrixData are overwritten with their default settings when the faceplates are updated. To prevent this, adjust these settings in @@SafetyMatrixTypicals.PDL within your project, or enter the parameters in file @@PCS7Typicals.CFG (which can be found in the OS server project path on the ES, under wincproj\<OSname>\WScripts).

The names of these attributes and properties can be found in the configuration dialog of the block icon, by selecting the block icon followed by "Configuration dialog..." in the context menu.



In file @@PCS7Typicals.CFG, you will need to extend the [Columns] and [Column00] sections. The name of the property is entered in [Columns] and the attribute in [Column00]. The sections should then appear as shown below (please note that they are case-sensitive).

Since the sections already contain entries for the properties and attributes, you must start with the first unused number (in the example: last number used = 10, next unused number = 11 = N), and increase the number for each entry.

[Columns]

(existing entries)

- ColumnN = SafetyGroupNumber
- TypeN = 3
- ColumnN+1 = SafetyGroupDescription
- TypeN+1 = 3
- ColumnN+2 = InitiatorLevel
- TypeN+2 = 3
- ColumnN+3 = ConfirmerLevel
- TypeN+3 = 3
- ColumnN+4 = CauseAckLevel
- TypeN+4 = 3
- ColumnN+5 = CauseBypLevel
- TypeN+5 = 3
- ColumnN+6 = CauseTagSimLevel
- TypeN+6 = 3
- ColumnN+7 = CauseClrFOLevel
- TypeN+7 = 3
- ColumnN+8 = EffectClrAlmLevel
- TypeN+8 = 3
- ColumnN+9 = EffectOvrLevel
- TypeN+9 = 3
- ColumnN+10 = EffectResLevel
- TypeN+10 = 3
- ColumnN+11 = EffectBypLevel

TypeN+11 = 3
 ColumnN+12 = EffectTagSimLevel
 TypeN+12 = 3
 ColumnN+13 = EffectClearLevel
 TypeN+13 = 3
 ColumnN+14 = DriversAckLevel
 TypeN+14 = 3

[Column00]

(existing entries)

[ColumnN]

Property0 = SafetyGroupNumber

[ColumnN+1]

Property0 = SafetyGroupDescription

[ColumnN+2]

Property0 = InitiatorLevel

[ColumnN+3]

Property0 = ConfirmerLevel

[ColumnN+4]

Property0 = CauseAckLevel

[ColumnN+5]

Property0 = CauseBypLevel

[ColumnN+6]

Property0 = CauseTagSimLevel

[ColumnN+7]

Property0 = CauseClrFOLevel

[ColumnN+8]

Property0 = EffectClrAlmLevel

[ColumnN+9]

Property0 = EffectOvrLevel

[ColumnN+10]

Property0 = EffectResLevel

[ColumnN+11]

Property0 = EffectBypLevel

[ColumnN+12]

Property0 = EffectTagSimLevel

[ColumnN+13]

Property0 = EffectClearLevel

[ColumnN+14]

Property0 = DriversAckLevel

Note

For more information on the structure of the @@PCS7Typicals.CFG file, refer to the WinCC Information System (Start > Simatic > WinCC > WinCC Information System in contents under Options > Options for Process Control > Graphic Object Update Wizard > Structure of the Configuration File).

5.6 Interconnections between the matrix and safety program

When establishing interconnections between the matrix and the remainder of the safety program, you must ensure that the interconnected blocks are all in the same shutdown group, or that the right communication blocks have been installed between the shutdown groups.

5.7 Using imported F channel drivers in the matrix

If the fail-safe signals are also to be used at a different point in the program, e.g. for display or interlocking purposes in the standard program, it is recommended that you create a process tag (CFC) for the signal and activate the "Use imported channel drivers (IEA support)" function in the matrix when transferring it to the project.

In the matrix, it is possible to use input signals that have already been used in the safety program.

External F channel drivers are F channel drivers that have not been placed by the Safety Matrix. To enable the Safety Matrix to interconnect external F channel drivers along with its own channel drivers, you must activate the "Use imported channel drivers (IEA support)" option when transferring the Safety Matrix to the project. This interconnects additional signals between the channel driver and the Safety Matrix. These are inputs "SIM_IN", "SIM_ON", and "SIM_V", which the matrix uses for the maintenance function, as well as "ACK_REI" and "ACK_REQ", used to signal an acknowledgment request in the Safety Matrix and perform acknowledgment from here.

To enable the Safety Matrix to integrate external F channel drivers into its full scope of functions, the "SIM_IN", "SIM_ON", "SIM_V", and "ACK_REI" I/Os may not be interconnected, as the Safety Matrix does not change the existing logic and thus does not interconnect the inputs.

Internal F channel drivers will be recreated whenever any changes affecting the "chart and parameters" are made; as part of this, any interconnections established previously will be lost. Do not configure any interconnections with F channel drivers in the driver chart of the Safety Matrix.

Changes, tracking changes, and acceptance

6.1 General information

Operational procedures

During maintenance work in or while making changes to a plant, the operational procedures must be followed at all times.

Before changes can be made to safety-related parts of the plant or functions, a risk assessment relating to the change itself and how it is to be made must be carried out, particularly if it is to be made during operation. Only make changes if a risk assessment has been carried out beforehand and ensure that change instructions resulting from this are followed.

An S7-400 F/FH system being operated as a safety system is protected by at least two passwords. Make sure you have the required access authorizations.

Note

You can find more information in Section Access protection (Page 13).

If changes are to be downloaded during operation, safety mode must be deactivated on the AS and then reactivated once downloading is complete.

Note

You can find more information in Section Safety mode and downloading the safety program (Page 99).

6.2 Preparing for changes

Offline/Online comparison

Before each CFC change, make sure that the offline and online programs are identical.

- To do this, open a chart from the AS in the CFC editor and switch to online mode.
- If there are differences between the offline and online versions, a message with an additional system prompt will appear; otherwise, the system will switch to online mode without prompting.

In the case of fail-safe functions in the AS, you must also check whether the signature from the last compilation matches the online signature in the AS (see Section Tracking changes in the safety program (Page 144)).

- If the signatures are not identical, or a prompt indicating that there are differences between the programs appears when you switch to online mode, you must determine what has changed and whether the changes can be downloaded.

Version Cross Manager

The "Version Cross Manager" can be used as a tool for comparing two project versions, as well as the current project with a backed-up version of it (see Section Tracking changes in the safety program (Page 144)).

- The comparison function in S7 F Systems can be used to list the differences in the safety program. You can access this comparison function by clicking the "Compare..." button in the "Edit Safety Program" window.
- The next window enables you to choose the versions to be compared.

Do not make any other changes if the offline and online versions are different and you do not know what the differences are.

CPU utilization

Before making any changes, check the CPU utilization with regard to memory and cycle time. You can query how much memory has been allocated using the "Module Information..." function in the CPU.

- To do this, open "Module Information..." in the CPU and check how much memory is allocated at present on the "Memory" tab.
- To determine the cycle utilization, blocks TIME_BEG and TIME_ENG must be installed in the OBs in use and interconnected with one another. In the relevant CFC, check the runtimes of the OBs at output TM_DIFF_TI of the respective TIME_END block. Please note that the runtimes of higher-priority OBs are also measured when this method of measuring the runtimes of lower-priority OBs is used, and will need to be subtracted accordingly.
With PCS 7 V7.0 SP1 and higher, and a S7-400H CPU with FW V4.5 and higher, the gross and net runtimes of the cyclic interrupt OBs can also be read at block CPU_RT.

Note

You can find more information in Section Cyclic interrupts (Page 18).

Checking the H parameters

In fault-tolerant systems, you must check the H parameters of the CPU.

- To do this, open the Object Properties of the CPU and switch to the "H Parameters" tab.
- If a "Cyclic interrupt OB with special handling" has been specified here, and the option "Use only calculated values" has been activated, use the "Calculate..." button to open the "Update Reserve: Calculation of Monitoring Times" window.
- Here, check the values in the entry fields "Runtime of the watchdog interrupt concerned [ms]" and "Work memory used for all data blocks in the user program [Kbyte]".
- Compare the runtime entered in the "Runtime of the watchdog interrupt concerned [ms]" field, for the cyclic interrupt OB with special handling, with the measured runtime (TIME_BEG, TIME_END of OB) or the runtime read at the CPU_RT block. The value entered should be approximately 10% greater than the measured one.
- Compare the value for "Work memory used for all data blocks in the user program [Kbyte]" with the object properties of the CPU block folder. On the "Blocks" tab there, you will find the amount of memory (in bytes) occupied by data blocks under "Size in work memory: Data". This value (expressed in Kbytes), plus some reserve, must be entered in the "Work memory used for all data blocks in the user program [Kbyte]" field.
- If the values in the entry fields do not match the actual values, you must enter the current values and determine the H parameters using the "Calculate..." button.

Note

You can find more information in Section Calculating monitoring times (Page 30).

Checking the hardware before downloading

Before downloading the changes - and in particular, before downloading hardware configuration changes with HCiR - make sure that no errors are present in the system.

- The HCiR procedure involves stopping a CPU, performing the download, and then restarting the CPU.
- Immediately after the transition to RUN Redundant, the other CPU is stopped and restarted, during which time it is updated to the current configuration.
- If there is an error in a redundant component (e.g. the interface module of the ET 200M), this can cause the connection to a PROFIBUS slave to fail and input/output modules to be passivated.

6.3 Changes in CFC

Changing the user program

Before you make changes to the standard or fail-safe program of an S7 F/FH systems, make sure that you can also download and check these changes. Changes to the standard or fail-safe program cannot be compiled or downloaded as part of a separate process. Therefore, you must follow the sequence below in full to make changes:

- Change CFC
- Check
- Compile
- Download
- Offline/online comparison
- Test
- Document

To change or add a fail-safe function, open the relevant CFC or add a new one.

New runtime group

When a new CFC is added, the system creates a new runtime group with the same name. Check whether its insert position in the editor is the position you require.

Empty F-runtime groups may arise when deleting safety functions. To remove these with PCS 7 V7.0, you need to delete them before compilation takes place. To do this, in the CFC editor select "Edit > Delete Empty Runtime Groups".

With PCS 7 V6.1, you need to activate the "Delete Empty Runtime Groups" option when carrying out compilation.

Before inserting blocks, establish the predecessor block. When inserting blocks, make sure that the sequence of the blocks corresponds to the signal flow. If necessary, change the block sequence in the runtime editor. Using the "Optimize Run Sequence" function for a fail-safe runtime group is not permitted.

Function test

To test your functions, you can use S7-PLCSIM or - depending on your PCS 7 version - you can download your program to a test AS. To perform testing with S7-PLCSIM, start S7-PLCSIM and download HW Config and the CFC charts. Depending on the version of S7-PLCSIM you are working with, you may need to enter either the password "plcsim" (up to PLCSIM V5.3) or the configured CPU password (PLCSIM V5.4 and higher) in order to download CFC charts. Testing with S7-PLCSIM is not a substitute for carrying out a function test in the plant.

6.4 Changes in HW Config

In HW Config, F signal modules may only be added or deleted. Changes to the configuration of a fail-safe module lead to passivation of the module by the self-test after the hardware configuration has been downloaded. If changes are required, check whether the signals are necessary for the process that is currently running. Establish (within the context of a risk assessment, if necessary) whether signal simulation is permissible.

Signature value of the signal module

Safety-related settings of the fail-safe signal modules are entered in the signature of the fail-safe program. The signature value of a fail-safe signal module appears as an input parameter on the F module driver.

To update this value, after safety-related parameters have been changed in the HW configuration the changes need to be compiled using "Generate module drivers" and downloaded in the CFC editor.

Safety information

In HW Config, you will receive a message when you make changes to safety-related parameters. If you are unable to or do not wish to finalize the changes for the reason specified above, you can cancel them here and close HW Config without saving.

6.5 Downloading changes/Complete downloading

Standard program

In an S7-400 F/FH system with a safety and standard program, changes can be made in the standard program in exactly the same way as with a standard system.

However, since the fail-safe program may also be affected by a program error in the standard program (e.g. an infinite loop in a user block), changes in the standard program should be treated in the same way as changes in the safety program.

Safety program

The procedure for downloading changes/complete downloading of programs with F blocks is identical to the one used for the standard program. Additionally, safety mode must be deactivated before downloading changes and then reactivated once this is complete.

Note

You can find more information in Section Safety mode and downloading the safety program (Page 99).

During a complete download, the CPU is stopped and must be restarted once the download is complete. When the CPU starts up, it goes into safety mode.

6.6 Tracking changes in the safety program

Backing up projects

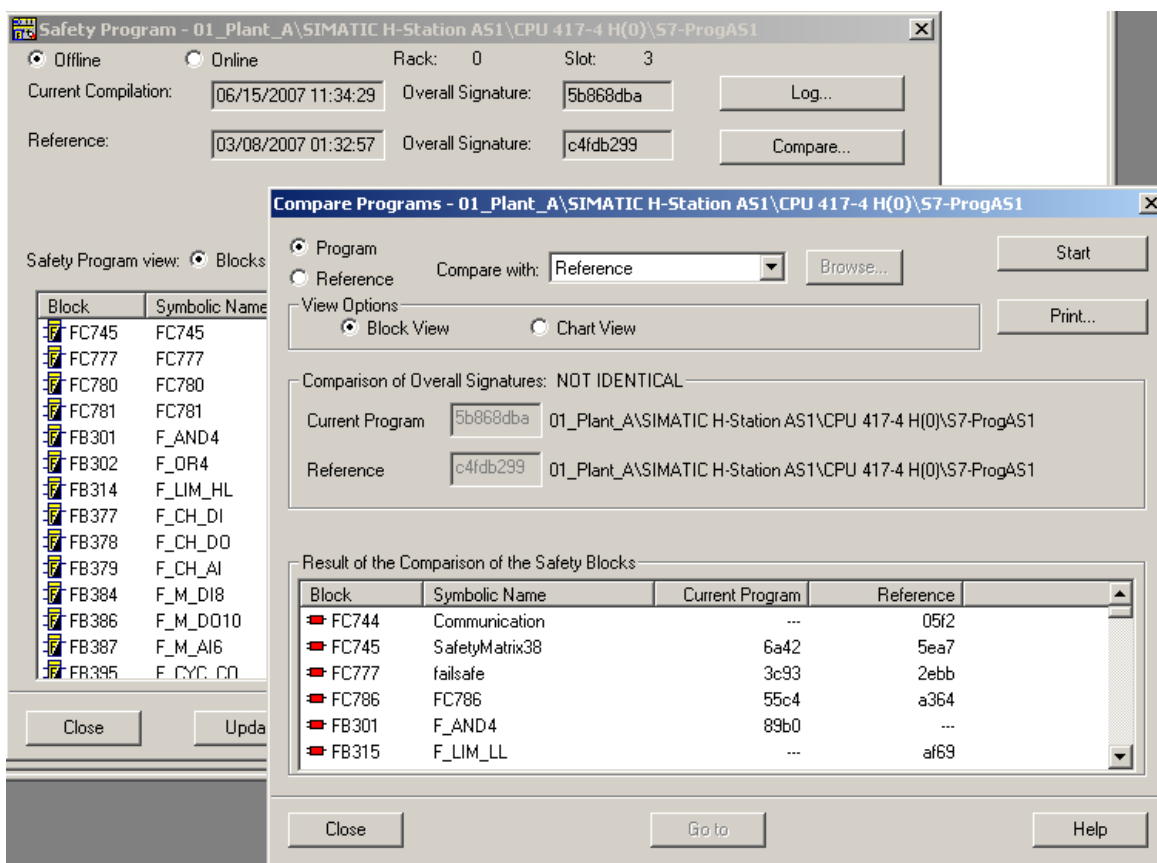
In order to track changes, it is necessary to back up the project at regular intervals both before and after changes are made. A plant-wide decision should be taken regarding where project backups are to be stored and how the individual versions are to be named.

Program comparison

The "Version Cross Manager" can be used for the entire program as a tool for comparing two program versions, as well as for comparing the current project with a backed-up version of it.

Options for comparing the safety program can be accessed by clicking the "Compare..." button in the "Edit Safety Program" dialog.

You can use this dialog to compare two safety programs and show and print the differences between them.



Using the "Compare with" drop-down list box, you can determine which programs you intend to compare.

- If you have activated the "Program" option button, you will be able to select one of the following programs here:

Reference	Last reference saved for this program
Last compilation	Last compilation of the program
Online	Program in the state currently downloaded on the F CPU
Another project	Any offline program, click ("Browse..." to select)

- If you have activated the "Reference" option button, you will be presented with the following selection options:

Current program	Current offline program
Last compilation	Last compilation of the program
Online	Program in the state currently downloaded on the F CPU
Another project	Any offline program, click ("Browse..." to select)

Signature of the safety program

The program signature is used to identify a safety program. It changes each time a change is made to the safety program. As a result, comparing the signature values of two safety programs provides an initial indication of what changes have been made in the safety program.

The signature of the current safety program in an AS can be found in the "Edit Safety Program" window. To do this, in the SIMATIC Manager select the CPU that contains the safety program and use the "Options > Edit Safety Program" menu command to open the relevant dialog. Here, you will see two signatures, each with a corresponding date:

- The most recent date on which the CFC charts with changes to the safety program were compiled, and the signature formed during this
- The date on which a safety program was saved as a reference, and which signature the reference program has

Save the program you have checked as a reference using the "Reference" button in the "Edit Safety Program" window; enter the password if requested to do so and answer "Yes" to the query that appears following this.

The signatures shown in the "Edit Safety Program" window do not need to match the signature in the AS. It may, for example, be possible that the most recent changes were compiled, but not downloaded and checked.

For this reason, you must also check the signature in the AS.

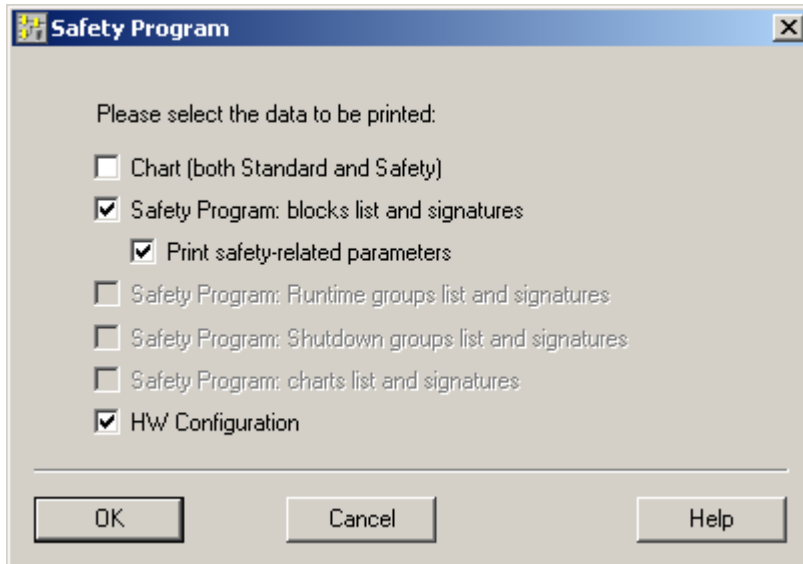
- To do this, go online in CFC and open the @F_Shtdn chart.
- While online, you can view the signature of the safety program in the AS at output F_SIG_OUT of the F_SHUTDOWN block.

For change tracking purposes, it is worth manually documenting both the signature value of the safety program and the name of the project backup in a log file.

6.7 Printing program data

You can use the "Edit Safety Program" dialog box to view, compare, and document many of the attributes of the safety program. It lists all the blocks that are contained in the safety program. F-runtime groups and charts can also be displayed.

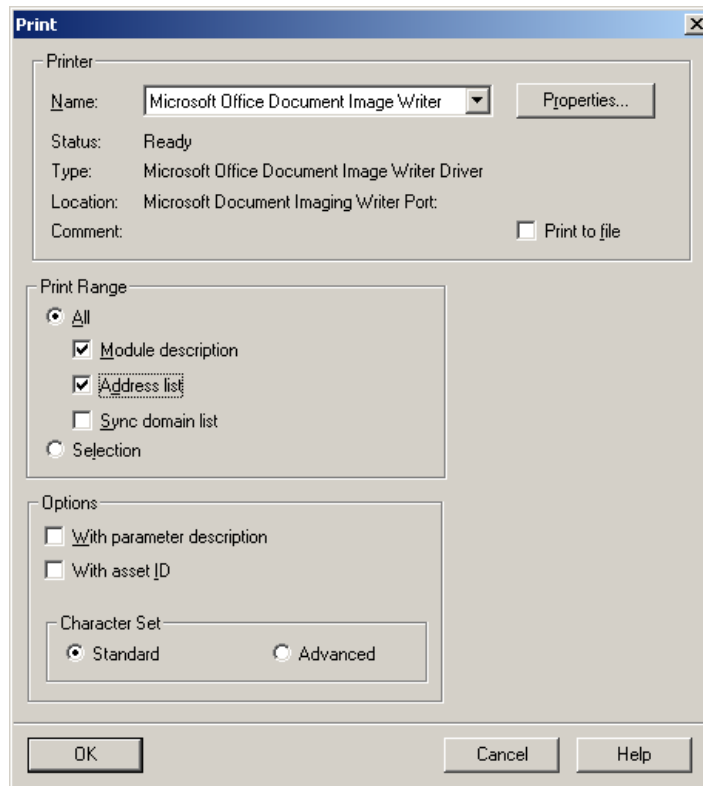
Here, you will also find options for documenting the current safety program:



If you want to print all the charts, activate the "Chart (standard and F chart)" option. Individual charts or all F charts can be printed from the SIMATIC Manager.

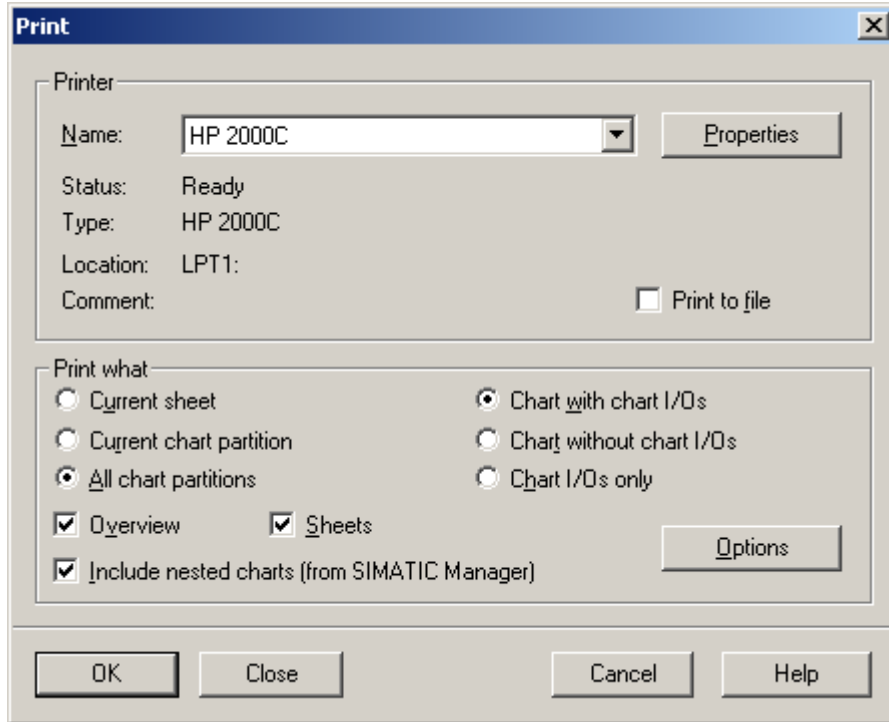
Activate the options shown above to print the safety program and the hardware.

Before the hardware is printed, another dialog appears:



Here, select the "All", "Module description", and "Address list" options.

To print individual or multiple charts from the SIMATIC Manager, you must make the CFC editor settings in the "Print" dialog. For this purpose, open a chart in CFC, go to the "Chart > Print" menu, and then select the printout options shown below.



In the chart directory of your AS, within the SIMATIC Manager you then have the option of highlighting charts for printing purposes, and printing the selected charts using the "File > Print > Object Content" menu command.

6.8 Plant acceptance

Note

You can find information on acceptance of a plant, changes, or F block types in the "S7 F/FH Systems Configuring and Programming" Manual.

(<http://support.automation.siemens.com/WW/view/en/2201072>)

Two checklists are available for system documentation purposes:

- "Process Safety – Configuration" checklist
This list is used for documenting the system design and the software and hardware settings of the automation system in use.
- "Process Safety – Verification/Testing"
This list enables you to document that the components in use have been tested against their requirements and specifications, and that the relevant function tests have been performed.

These lists are only templates and may be adapted to the requirements of your plant at any time.