SIEMENS

SIMATIC

S7-400H Programmable Controller Fault-Tolerant Systems

Manual

This manual has the order number:

6ES7988-8HA10-8BA0

Edition 07/2000
A5E00068197-04

Safety Guidelines

This manual contains notices which you should observe to ensure your own personal safety, as well as to protect the product and connected equipment. These notices are highlighted in the manual by a warning triangle and are marked as follows according to the level of danger:

Danger

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

Warning

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

Caution

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

Note

draws your attention to particularly important information on the product, handling the product, or to a particular part of the documentation.

Qualified Personnel

Only **qualified personnel** should be allowed to install and work on this equipment. Qualified persons are defined as persons who are authorized to commission, to ground, and to tag circuits, equipment, and systems in accordance with established safety practices and standards.

Correct Usage

Note the following:

Warning

This device and its components may only be used for the applications described in the catalog or the technical descriptions, and only in connection with devices or components from other manufacturers which have been approved or recommended by Siemens.

This product can only function correctly and safely if it is transported, stored, set up, and installed correctly, and operated and maintained as recommended.

Trademarks

SIMATIC®, SIMATIC HMI® and SIMATIC NET® are registered trademarks of SIEMENS AG.

Some of other designations used in these documents are also registered trademarks; the owner's rights may be violated if they are used by third parties for their own purposes.

Siemens AG
Bereich Automatisierungs- und Antriebstechnik
Geschaeftsgebiet Industrie-Automatisierungssysteme
Postfach 4848, D- 90327 Nuernberg

Disclaimer of Liability

We have checked the contents of this manual for agreement with the hardware and software described. Since deviations cannot be precluded entirely, we cannot guarantee full agreement. However, the data in this manual are reviewed regularly and any necessary corrections included in subsequent editions. Suggestions for improvement are welcomed.

Important Notes

Purpose of the manual

The present manual is intended for persons involved in the areas of configuration, commissioning and servicing of programmable logic control systems.

To help you get familiar with the product, we recommend that you start with the example in Chapter 3. It shows you an easy method of getting started on the subject of fault-tolerant systems.

Basic knowledge required

In order to understand the manual, you will need to be familiar with the general principles of automation technology.

Knowledge of S7 programs is also a prerequisite; you can read more about S7 programs in the *Programming with STEP 7* manual. As you need the STEP 7 standard software while you are configuring, you should also be familiar with running the standard software, as explained in the STEP 7 User Manual.

Validity of the manual

The manual is valid for CPU 417-4H firmware version V2.1.0 or higher and option package S7 H Systems, version V5.1 or higher.

Online Help

In addition to the manual, detailed support on how to use the software is provided by the online Help system integrated in the software.

The Help system can be accessed using a number of interfaces:

- In the **Help** menu are a number of commands: **Contents** opens the Help index. You will find help on fault-tolerant systems at **Call Help on options packages, configuring fault-tolerant systems**.
- **How to Use Help** provides detailed instructions on how to use online Help.
- Context-sensitive Help provides information on the current context – for example, on an open dialog box or an active window. It is accessed by means of the “Help” button or F1.
- Another form of context-sensitive Help is the status bar. A brief explanation of each menu command is displayed here when you point the mouse pointer at the menu command.
- A brief explanation of the toolbar buttons is also shown when the mouse pointer comes to rest for a short time on the buttons.

If you would like to read information from online Help in printed form, you can print individual topics, books or the entire Help.

Feedback on documentation

We need your help to enable us to provide you and future users with optimum documentation. Should you have any remarks on this *manual* or *online Help*, please fill out the remarks form at the end of the manual and return it to the address shown on the form. Please also indicate your personal opinion of the manual.

SIMATIC Training Center

We offer a number of courses to help you become familiar with the SIMATIC S7 programmable logic controller. Please contact your regional training center or the central training center in D-90327 Nuremberg.
Phone: +49 (911) 895-3200.

Further support

The Nuremberg H/F Competence Center holds a special workshop on fault-tolerant SIMATIC S7 programmable logic controllers. In addition, the H/F Competence Center will provide on-site assistance with configuration, commissioning and other problems.

Further information can be obtained as follows:
Phone: +49 (911) 895-4759
Fax: +49 (911) 895-4519

SIMATIC Customer Support Hotline

Available 24 hours a day, worldwide:

SIMATIC Hotline

Worldwide (Nuremberg) **Technical Support** (FreeContact) Local time: Mon. through Fri. 7.00 a.m. to 5.00 p.m. Phone: +49 (180) 5050-222 Fax: +49 (180) 5050-223 E-mail: techsupport@ad.siemens.de GMT: +1:00	**Worldwide** (Nuremberg) **Technical Support** (subject to charge, with SIMATIC Card only) Local time: Mon. through Fri. 0.00 a.m. to 12.00 p.m. Phone: +49 (911) 895-7777 Fax: +49 (911) 895-7001 GMT: +01.00	
Europe / Africa (Nuremberg) **Authorization** Local time: Mon. through Fri. 7.00 a.m. to 5.00 p.m. Phone: +49 (911) 895-7200 Fax: +49 (911) 895-7201 E-mail: authorization@nbgm.siemens.de GMT: +1:00	**America** (Johnson City) **Technical Support and Authorization** Local time: Mon. through Fri. 8 a.m. to 5.00 p.m. Phone: +1 423 461-2522 Fax: +1 423 461-2289 E-mail: simatic.hotline@sea.siemens.com GMT: –5:00	**Asia/Australia** (Singapore) **Technical Support and Authorization** Local time: Mon. through Fri. 8.30 a.m. to 5.30 p.m. Phone: +65 740-7000 Fax: +65 740-7001 E-mail: simatic.hotline@sae.siemens.com.sg GMT: +8:00
The SIMATIC Hotline languages are normally German and English; French, Italian and Spanish are also spoken on the Authorization Hotline.		

SIMATIC Customer Support Online Services

SIMATIC Customer Support provides you with comprehensive additional information in SIMATIC products by means of its online services:

- You can obtain up–to–date information
  - on the **Internet** at `http://www.ad.siemens.de/simatic`
- Current product information leaflets and downloads which you may find useful for your product:
  - on the **Internet** at `http://www.ad.siemens.de/simatic–cs`
  - From our **Bulletin Board System** (BBS) in Nuremberg (*SIMATIC Customer Support Mailbox)* by dialing +49 (911) 895-7100.

    To dial the mailbox, use a modem having up to V.34 (28.8 kbps) and set its parameters as follows: 8, N, 1, ANSI, or dial in using ISDN (x.75, 64 kbps).
- You can find your local point–of–contact for Automation & Drives in our contacts database
  - on the **Internet** at `http://www3.ad.siemens.de/partner/search.asp`

Contents

Fault-Tolerant Systems in Automation Engineering

1

This chapter contains an introduction to redundant and fault-tolerant programmable logic controllers.

1.1 Redundant Programmable Logic Controllers in the SIMATIC Series

Economic, and thus resource-sparing and low-pollution production can be achieved nowadays in all branches of industry only by employing a high degree of automation. At the same time there is a demand for fail-safe programmable logic controllers with the greatest degree of distribution possible.

Redundant programmable logic controllers from Siemens have proved themselves in operation and thousands are in service.

Perhaps you are already familiar with one of the fault-tolerant systems such as the SIMATIC S5-115H and S5-155H, or the fail-safe S5-95F and S5-115F systems.

The S7-400H is the latest fault-tolerant PLC and we will be presenting it on the pages that follow. It is a member of the SIMATIC S7 system family, meaning that you can fully avail yourself of all the advantages of the SIMATIC S7.

Operating objectives of Redundant PLCs

Redundant programmable logic controllers are used in practice with the aim of achieving a higher degree of availability or fault tolerance.

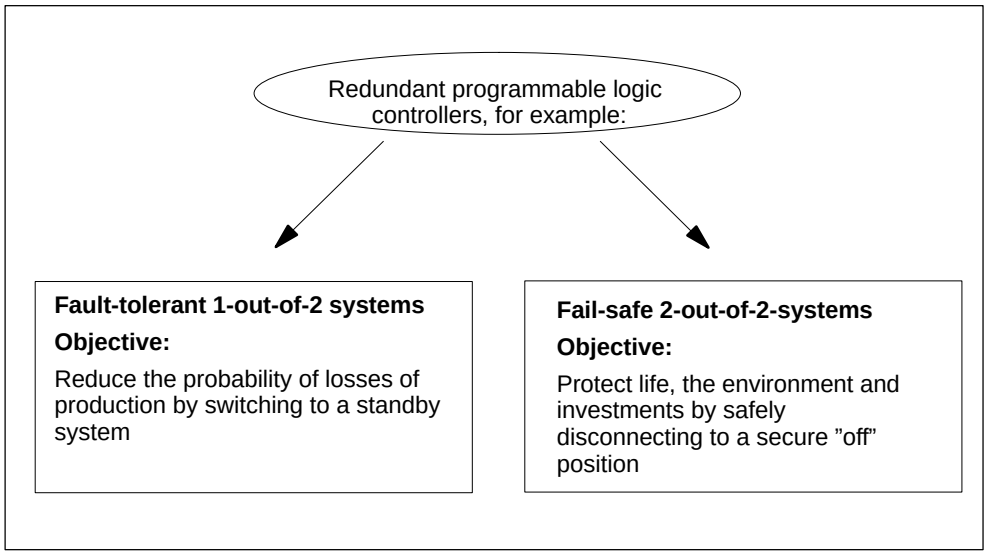


Figure 1-1 Operating Objectives of Redundant Programmable Logic Controllers

Note the difference between fault-tolerant systems and fail-safe systems. The S7-400H is a fault-tolerant programmable logic controller that can be used only with additional means for controlling processes relevant to safety.

Why do we have fault-tolerant programmable logic controllers?

The objective of using high-availabilty programmable logic controllers is a reduction of losses of production. It does not matter whether the losses are caused by an error or as a result of maintenance work.

The higher the costs of a stoppage, the more worthwhile it is to use a fault-tolerant system. The generally higher investment costs of fault-tolerant systems are quickly compensated by the avoidance of losses of production.

Software redundancy

In a large number of applications, requirements in respect of the quality of redundancy or the number of system sections that necessitate redundant PLCs are not high enough to warrant the use of a specific fault-tolerant system. Frequently, simple software mechanisms are sufficient to allow continuation of a failed control task on a substitute system in the event of an error.

The “SIMATIC S7 Software Redundancy” options software can run on S7-300 and S7-400 standard systems to control processes that tolerate transfer times to a substitute system within seconds, such as water works, water treatment systems or traffic flows.

1.2 Increasing System Availability

The S7-400H programmable logic controller meets these high requirements for availability, intelligence and distribution that are required of state-of-the-art programmable logic controllers. Further, it features all the functions for acquiring and preparing process data and for controlling, regulating and monitoring units and systems.

System-wide universality

The S7-400H programmable logic controller and all other SIMATIC components, such as the SIMATIC PCS7 control system, are harmonized. Total system universality, from the supervisory console to the sensors and actuators, is a matter of course and guarantees maximum system performance.

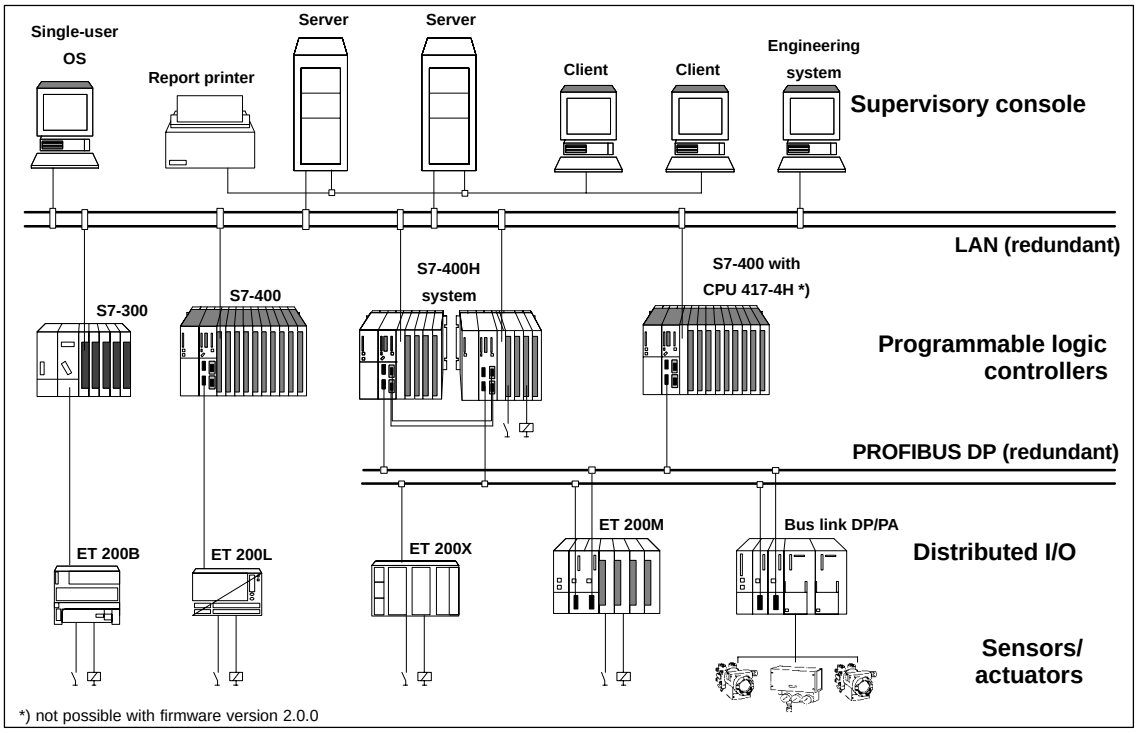


Figure 1-2 Universal Automation Solutions with SIMATIC

Graduated availability by duplicating components

The S7-400H is designed with redundancy so that it remains available at all events. This means that all major components are duplicated.

The components that are duplicated as a matter of policy are the central processing unit (CPU), the power supply and the hardware for interconnecting the two central processing units.

You can decide for yourself whether you wish to duplicate more components for the process you are going to automate and thus enhance their availability.

Redundant nodes

Redundant nodes represent the fault tolerance of systems with redundant components. The independence of a redundant node is given when the failure of a component within the node does not result in reliability constraints in other nodes or in the entire system.

The availability of the entire system can be illustrated in a simple manner by means of a block diagram. With a 1-out-of-2 system, **one** component of the redundant node may fail without impairing the operability of the overall system. The weakest link in the chain of redundant nodes determines the availability of the overall system.

Without malfunction (Figure 1-3).

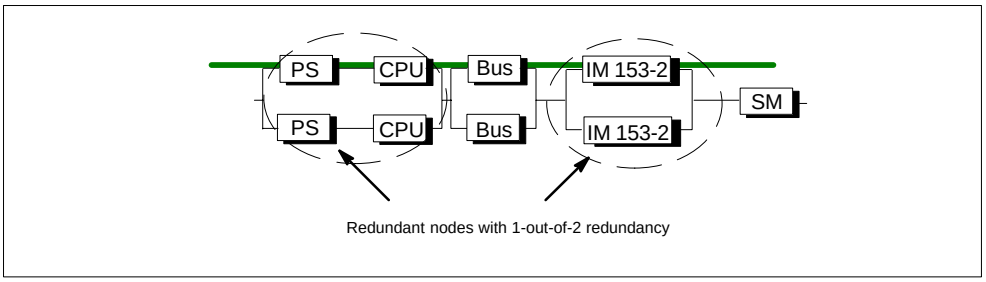


Figure 1-3 Example of Redundancy in a Network without Malfunction

With malfunction

In Figure 1-4, one component may fail per redundant node without the functionality of the overall system being impaired.

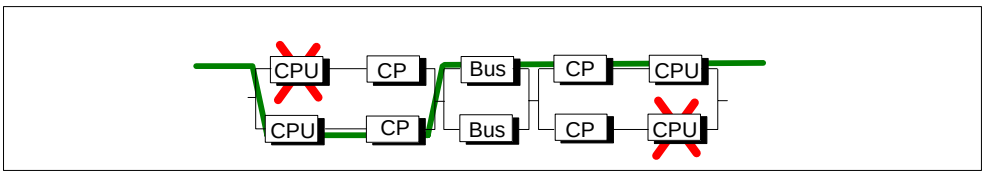


Figure 1-4 Example of Redundancy in a 1-out-of-2 System with Malfunction

Failure of a redundant node (total failure)

In Figure 1-5, the entire system is no longer operable since both subcomponents have failed in a 1-out-of-2 redundant node (total failure).

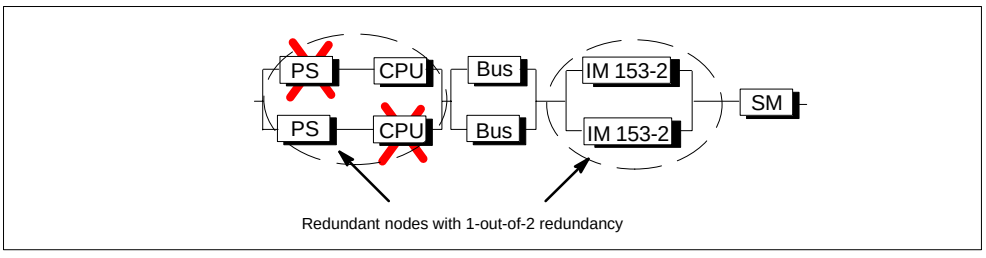


Figure 1-5 Example of Redundancy in a 1-out-of-2 System with Total Failure

S7-400H Installation Options

2

The first part of the description starts with the basic configuration of the fault-tolerant S7-400H programmable controller and the components making up the S7-400H base system. We then describe the hardware components with which you can expand this base system.

The second part describes the software applications with which you can configure and program the S7-400H. In addition, a description is given of the additions and extensions, compared to the S7-400 standard system, that you will require for programming your user program in order to be able to react specifically to the properties of the S7-400H that enhance availability.

Figure 2-1 shows an example of the configuration of an S7-400H with common distributed I/O and a connection to a redundant system bus. On the next few pages we will describe step by step the hardware and software components necessary for configuring and operating the S7-400H.

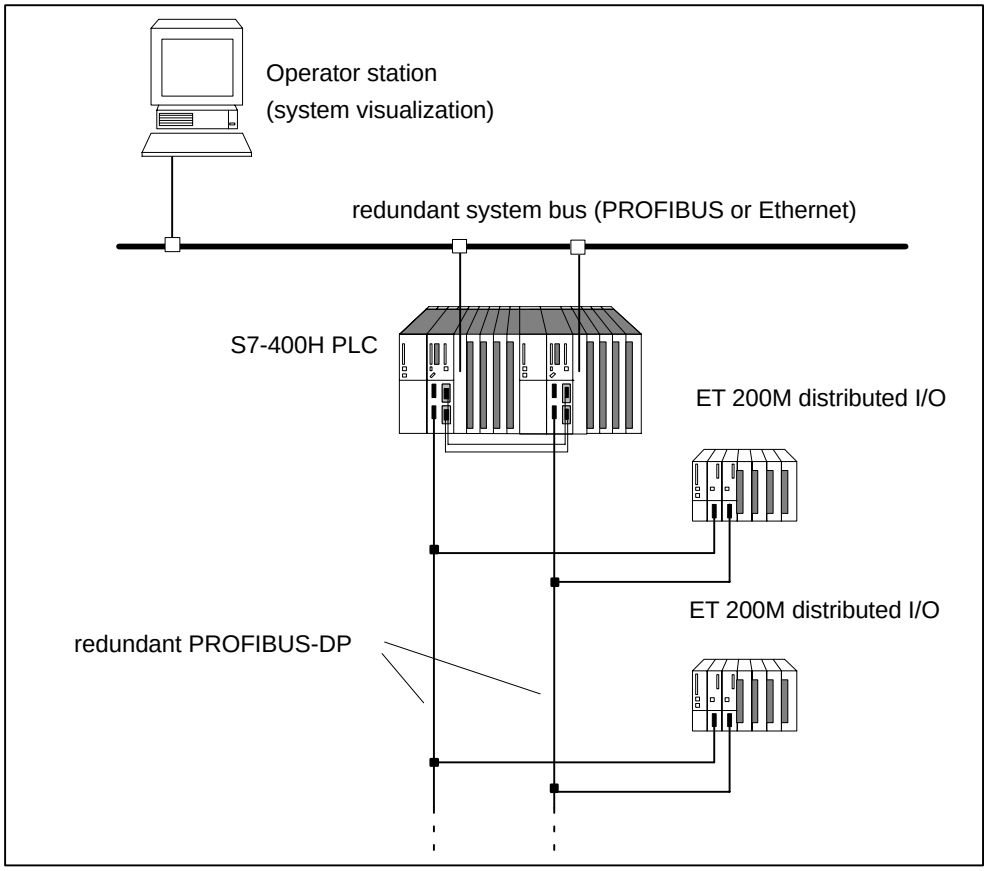


Figure 2-1 Overview

Further information

The components of the S7-400 standard system are also used in the fault-tolerant S7-400H programmable logic controller. A detailed description of all the hardware components of the S7-400 and S7-400H may be found in the Reference Manual *S7-400, M7-400 Programmable Controllers, Module Specifications*.

The same rules as for a standard S7-400 system apply to designing the user program and the usage of blocks for the fault-tolerant S7-400H programmable logic controller. Please take note of the descriptions in the *Programming with STEP 7* manual and in the *System Software for S7-300/400, System and Standard Functions* Reference Manual.

2.1 Base System of the S7-400H

Hardware of the Base System

By base system of the S7-400H we mean the minimum configuration of the S7-400H. The base system consists of all the requisite hardware components that make up the fault-tolerant control system. Figure 2-2 shows the components in the installation.

You can upgrade the base system by means of standard modules from the S7-400. There are restrictions in the case of the function and communication processors (see appendix E).

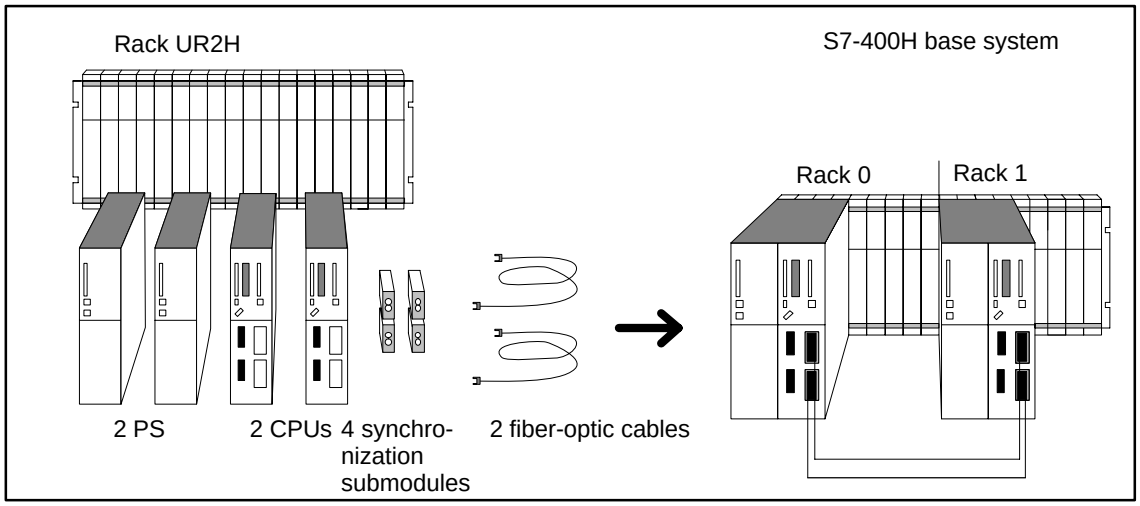


Figure 2-2 Hardware of the S7-400H Base System

CPU 417-4 H central processing unit

At the heart of the S7-400H are the two central processing units CPU 417-4H. Setting of the synchronization submodules, which have to be plugged into the CPU, defines the rack numbers. In the following we will refer to the CPU in rack 0 as CPU 0,and to the CPU in rack 1 as CPU 1.

Mounting rack for S7-400H

We recommend you the UR2-H mounting rack for the S7-400H. The mounting rack makes it possible to configure two separate subsystems, each containing nine slots, and is suitable for installation in 19” cabinets.

Alternatively, you can also configure the S7-400H on two separate mounting racks. Two mounting racks, the UR1 and UR2, are available for this purpose.

Power supply

As a power supply, you will require for each CPU 417-4 H – or, to be more precise, for each of the two subsystems of the S7-400H – a power supply module from the standard range of the S7-400.

Power supply modules for rated input voltages of 24 V DC and 120/230 V AC are available with input powers of 4, 10 and 20 A.

To enhance the availability of the power supply, you can also use two redundant power supplies in each subsystem. In this case you should use the PS 407 10 A R power supply module for rated voltages of 120/230 V AC with an output power of 10 A.

Synchronization submodules

The synchronization submodules are used to connect the two central processing units. They are installed in the central processing units and interconnected by means of fiber-optic cables.

Two synchronization submodules have to be inserted in each CPU.

Fiber-optic cables

The fiber-optic cables are inserted into the synchronization submodules and form the physical connection (redundant link) between the two central processing units.

You will find further information on handling and adjusting the synchronization submodules and the fiber-optic cables in the *S7-400, M7-400 Programmable Controllers, Module Specifications* Reference Manual and in the *S7-400, M7-400 Programmable Controllers, Hardware and Installation* Manual.

2.2 I/O for the S7-400H

For the S7-400H you can use virtually any of the input/output modules featured in the SIMATIC S7 system range. The I/O can be used in

- central racks
- expansion units
- distributed over PROFIBUS DP.

The function modules (FMs) and communication processors (CPs) that can be used in the S7-400H will be found in Appendix E.

I/O configuration versions

In addition to the power supplies and central processing units that are always used as redundant modules, there are the following configuration versions for the input/output modules:

- Single-channel, one-sided configuration with normal availability

  With the single-channel, one-sided configuration single input/output modules are present (single-channel). The input/output modules are located in just one of the subsystems and are only addressed by that subsystem.

- Single-channel, switched configuration with enhanced availability

  With the single-channel switched (distributed) configuration single input/output modules are present (single-channel) but can be addressed by either subsystem.

Further information

You will find detailed information on the usage of I/O in Chapter 6.

2.3 Communication

For communication tasks on the S7-400H you can use almost any communications components offered in the SIMATIC system range.

This applies to communication components used either with central I/O or distributed I/O such as

- system buses (Industrial Ethernet, PROFIBUS)
- point-to-point connection

Availability of communications

You can vary the availability of communications with the S7-400H. There are different solutions for the S7-400H in keeping with your communication requirements. They range from a simple linear network structure to a redundant optical two-fiber loop.

Fault-tolerant communication over PROFIBUS or Industrial Ethernet is supported entirely with S7 communication functions.

Programming and configuration

Apart from the use of additional hardware components, there are basically no differences with regard to configuration and programming compared to standard systems. Fault-tolerant connections have to be configured only; specific programming is not necessary.

All communication functions required for operating fault-tolerant communications have been integrated in the operating system of CPU 417-4H and run automatically and in the background – for example, monitoring of the communication connection or automatic switching to a redundant connection in the event of a malfunction.

Further information

You will find detailed information on the subject of communications with the S7-400H in Chapter 7.

2.4 Configuration and Programming Applications

The S7-400H is configured and programmed with STEP 7 just like any other SIMATIC S7 programmable logic controller.

After configuration with STEP 7, you treat the S7-400H as a normal S7-400 system.

For you this means that you can use your full knowledge of the SIMATIC S7 and, for example, only have to take minor constraints into account when writing your user program. However, there are also H-specific additions to the configuration. Redundant components are monitored by the operating system, which independently performs switching in the event of a fault. You have already configured the information required for this in STEP 7 and it is known to the system.

You will find detailed information on this subject in online Help and in Chapter 8.

Requisite software

The software components specified in Section 8.1 are required for configuration and programming.

Optional software

All standard tools, engineering tools and runtime software that can be used on the S7-400 can, of course, also be used on the S7-400H.

2.5 User Program

The rules applicable to the design and programming of the standard S7-400 system apply similarly to the S7-400H.

The user programs are stored in an identical form in the two central processing units and are executed simultaneously (event-synchronous).

From the viewpoint of user program execution, the S7-400H behaves in exactly the same manner as a standard system. The synchronization functions are integrated in the operating system and run automatically and totally in the background. There is no need to take these functions into account in the user program.

In order to be able to react to the lengthening of the cycle time due to updating, for example, a few specific blocks allow you to optimize your user program in this respect.

Specific organization blocks and system functions for the S7-400H

Apart from the blocks that can be used on both the S7-400 and the S7-400H, there are further additional blocks for the S7-400H with which you can influence the redundancy functions.

You can react to redundancy errors of the S7-400H with the following organization blocks:

- OB 70, I/O redundancy errors
- OB 72, CPU redundancy errors

Using system function SFC 90 “H_CTRL” you can inhibit and re-enable link-up and updates of the 417-4H CPUs. You can also affect the scope and execution of the cyclical self-test.

Note

With a fail-safe system, the periodic self-tests must not be inhibited and then enabled again.
For more details refer to the manual *S7-400F and S7-400FH Programmable Controllers; Fail-Safe Systems.*

Further information

You will find detailed information on the programming of the above-mentioned blocks in the manual called *Programming with STEP 7* and in the Reference Manual called *System Software for S7-300/400, System and Standard Functions*.

2.6 Documentation

The following illustration presents an overview of the description of the different components and possibilities presented by the S7-400H PLC.

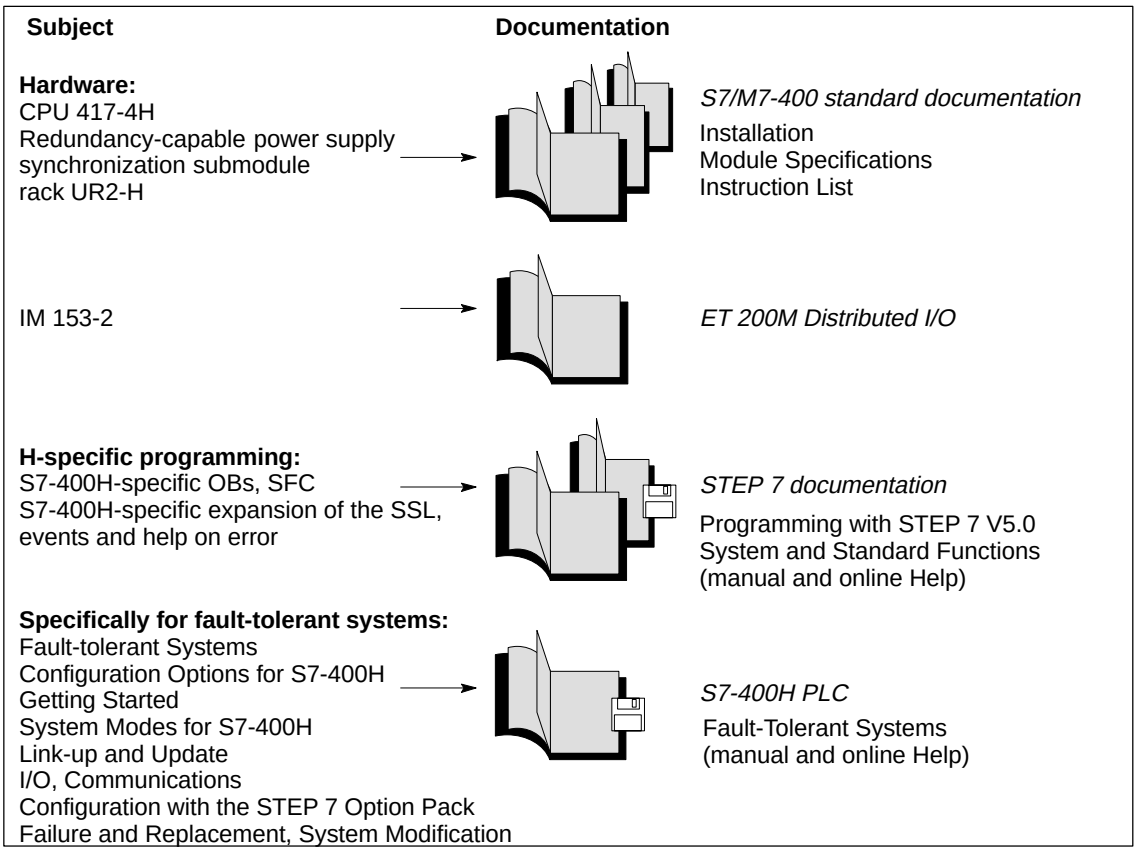


Figure 2-3 User Documentation for Fault-Tolerant Systems

Note

You will find the manuals listed in Figure 2-3 on the S7-400H product CD.

Getting Started

3

This guide walks you through the steps that have to be performed to commission the system by means of a specific example and results in a working application. You will learn how an S7-400H programmable logic controller operates and become familiar with its response in the event of a fault.

It takes about one to two hours to work through this example, depending on your previous experience.

3.1 Requirements

The following requirements must be met:

A permitted version of the STEP 7 standard software and the ”S7 Fault-Tolerant System” option pack are correctly installed on your programming device (refer to Section 8.1).

You must have the modules required for the hardware configuration:

- an S7-400H PLC consisting of:
  - 1 mounting rack, UR2-H
  - 2 power supplies, PS 407 10A
  - 2 CPU 417-4 H
  - 4 synchronization submodules
  - 2 fiber-optic cables
- an ET 200M distributed I/O device having an active backplane bus with
  - 2 IM 153-2
  - 1 digital input module, SM321 DI 16 x DC24V
  - 1 digital output module, SM322 DO 16 x DC24V
- the necessary accessories such as PROFIBUS shielded cables, etc.

3.2 Configuring Hardware and Starting Up the S7-400H

Installing Hardware

To configure the S7-400H as illustrated in Figure 3-1, perform the following steps:

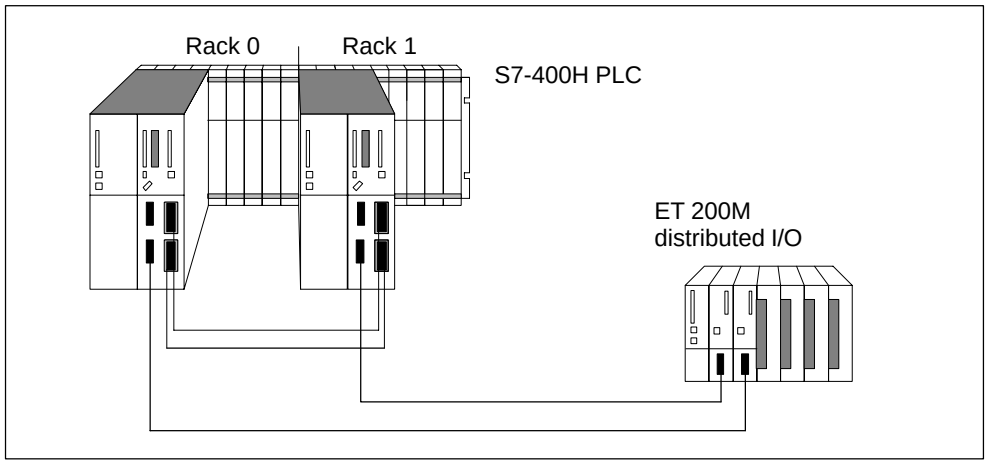


Figure 3-1 Hardware Configuration

1. Configure the two subunits of the S7-400H PLC as described in the *S7-400, M7-400 Programmable Controllers, Hardware and Installation/Module Specifications* manuals. In addition, you must:
   - Set the mounting rack number by means of the switches on the synchronization submodules. The setting is applied by the CPU after POWER ON and a subsequent memory reset by means of the mode selector. If the mounting rack number is not set correctly you will not have online access and the CPU will not run in certain circumstances.
   - Insert the synchronization submodules into the CPUs. Then screw up the additional front bezels to activate them (refer to *S7-400, M7-400 Programmable Controllers, Hardware and Installation*).
   - Connect the fiber-optic cables (always connect the two upper synchronization submodules and the two lower synchronization submodules of the CPUs). Lay the fiber-optic cable so that it is protected from any damage.

     Make sure with the route wires in addition that the two fiber-optic cables are always laid so that they are isolated from each other. Laying them separately enhances their availability and protects then from potential dual faults in the event, say, of simultaneous interruption of the fiber-optic cables.

     In addition, make sure that the fiber-optic cables are plugged into the two CPUs before turning on the power supply or turning on the system. If they are not, the two CPUs might both process the user program as master CPUs.
2. Configure the distributed I/O as described in the *ET 200M Distributed I/O Device* manual.

3. Connect the programming device to the first CPU 417-4 H (CPU0). This CPU should be the master CPU of the S7-400H.
4. A high-quality RAM test is performed after power on. It requires approximately 8 seconds per megabyte of RAM. During this time the CPU cannot be addressed via the multipoint interface and the STOP LED flashes. If there is a backup battery, the test will not be performed on further POWER ONs.
5. Perform a memory reset for both CPUs using the mode selector. This applies the set mounting rack numbers of the synchronization modules to the operating system of the CPU.
6. Perform commissioning individually for each CPU as described in the *S7-400, M7-400 Programmable Controllers, Hardware and Installation* manual. After loading the program carry out a warm restart: first for the CPU you want as the master CPU, and then for the standby CPU.
7. Switch the two CPUs of the S7-400H to STOP.

Starting up the S7-400H

To start up the S7-400H, perform the following steps:

1. Open the “HProject” in SIMATIC Manager. The configuration is the same as the hardware configuration described in “Requirements”.
2. Open the hardware configuration of the project by selecting the “Hardware” object and execute the pop-up menu command **Object > Open** with the right mouse button. When you have an identical configuration, you can proceed with step 6.
3. If your hardware configuration is different from that of the project – for example, the module types, MPI addresses or DP address – you must adjust and save the project accordingly. You will find descriptions in the basic help for SIMATIC Manager.
4. Open the user program in the “S7 program” folder.

   The “S7 program” folder is assigned only to CPU0 in the offline view. The user program can run on the hardware configuration described. It makes the LEDs on the digital output module light up in the form of a running light.
5. If necessary, modify the user program – to adapt it to your hardware configuration, for example – and save it.
6. Load the user program into CPU0 with the menu command **PLC > Load**.
7. Start the S7-400H PLC by switching the mode selector, first for CPU0 and then for CPU1, to RUN-P.

   Result: CPU0 starts up as the master CPU and CPU1 as the standby CPU. After the link-up and update of the standby CPU the S7-400H switches to the Redundant system mode and executes the user program (run light on digital output module).

Note

You can start and stop the S7-400H programmable logic controller using the programming device too. You will find more information on this in online Help of the S7-400H options package.

3.3 Examples of Fault-Tolerant System Response in the Event of Faults

Example 1: Failure of a central processing unit or power supply

Initial situation: The S7-400H is in the Redundant system mode.

1. Cause CPU0 to fail by turning off the power supply.

   Result: The LEDs REDF, IFM1F and IFM2F light on CPU1. CPU1 goes to Solo mode and the user program continues to run.

2. Turn the power supply back on.

   Result:

   – CPU0 performs an automatic LINK-UP and UPDATE.

   – CPU0 changes to RUN and now operates as the standby CPU.

   – The S7-400H is now in the Redundant system mode.

Example 2: Failure of a fiber-optic cable

Initial situation: The S7-400H is in the Redundant system mode. The mode selector of each CPU is at the RUN or RUN-P position.

1. Disconnect one of the fiber-optic cables.

   Result: The LEDs REDF and IFM1F or IFM2F (depending on which fiber-optic cable was disconnected) now light on the two CPUs. The original master CPU (CPU0) changes to Solo mode and the user program continues to run.

2. Reconnect the fiber-optic cable that you disconnected earlier.

3. Restart the original standby CPU (CPU1), which is now at STOP, by means of STEP7 “operating status”, for example.

   Result:

   – CPU1 performs an automatic LINK-UP and UPDATE.

   – The S7-400H reverts to the Redundant system mode.

System and Operating Modes of the S7-400H

4

This chapter features an introduction to the subject of S7-400H fault-tolerant systems.

You will learn the basic concepts that are used in describing how fault-tolerant systems operate.

Following that, you will receive information on fault-tolerant system modes. These modes depend on the operating modes of the different fault-tolerant CPUs, which will be described in the section that follows after that one.

In describing these operating modes, this section concentrates on the behavior that differs from a standard CPU. You will find a description of the normal behavior of a CPU in the corresponding operating mode in the *Programming with STEP 7 manual.*

The final section provides details on the modified time response of fault-tolerant CPUs.

4.1 Introduction

The S7-400H consists of two redundant configured subsystems that are synchronized via fiber-optic cables.

The two subsystems create a fault-tolerant programmable logic controller operating with a two-channel (1-out-of-2) structure on the “active redundancy” principle.

What does active redundancy mean?

Active redundancy, frequently referred to as functional redundancy too, means that all redundant resources are constantly in operation and are simultaneously involved in the execution of the control task.

This means for the S7-400H that the user program in the two CPUs is completely identical and is executed simultaneously (synchronously) by the two CPUs.

Declaration

To identify the two subsystems, we use the traditional expressions of “master” and “standby” for two-channel fault-tolerant systems in this description. The standby always operates so that it is synchronized with the events on the master, meaning that it does not wait for an error instance.

A distinction between the master CPU and the standby CPU is primarily important for ensuring reproducible error reactions. Thus, for example, the standby CPU switches to STOP in the event of the redundant link failing, whereas the master CPU remains at RUN. Further, unlike the standby CPU, the master CPU can access a switched I/O in the STOP mode.

Master/standby assignment

When the S7-400H is turned on for the first time, the first CPU to be stated up becomes the master CPU; the other CPU becomes the standby CPU.

Once the master/standby assignment has been established, it remains like that upon simultaneous POWER ON.

The master/standby assignment is modified by:

1. The standby CPU starting before the master CPU (interval of at least 3 s)
2. Failure or STOP of the master CPU in the Redundant system mode
3. No fault was found in ERROR-SEARCH mode (refer also to Section 4.3.6)

Synchronizing the subsystems

The master and standby CPUs are linked by means of fiber-optic cables. The two CPUs maintain event-driven synchronous program scanning over this link.

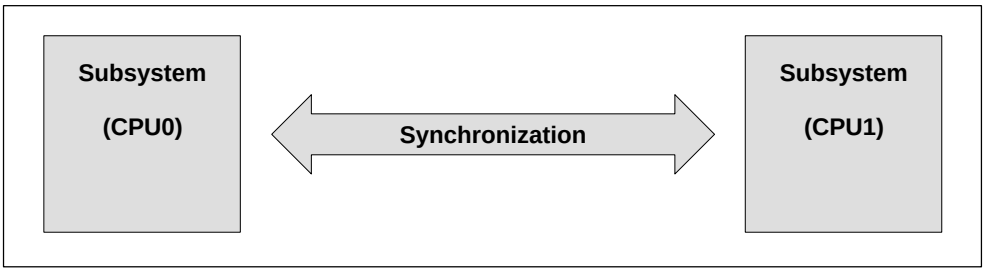


Figure 4-1 Synchronizing the Subsystems

Synchronization is performed automatically by the operating system and has no effect on the user program. You create your program in the manner in which you are accustomed for standard CPUs on the S7-400.

Event-driven synchronization procedure

The “event-driven synchronization” procedure patented by Siemens has been used on the S7-400H. This procedure has proved itself in practice and has already been used for the S5-115H and S5-155H PLCs.

Event-driven synchronization means that data synchronization between the master and the standby is performed for any event which may result in a different internal mode of the subsystems.

The master and standby CPUs are synchronized upon:

- direct access to the I/O
- interrupts
- updating of user times – for example, S7 timers
- modification of data by communication functions

Continued operation – free from discontinuities – even in the event of loss of redundancy of a CPU

The event-driven synchronization procedure ensures continued operation at all times, and free from discontinuities, by the standby CPU even in the event of a master CPU failure.

Self-tests

Malfunctions have to be detected, isolated and reported as quickly as possible. Consequently, widely-ranging self-test functions have been implemented on the S7-400H and run automatically and entirely in the background after POWER ON, in cyclic operation and even in the ERROR-SEARCH mode.

The following components and functions are tested:

- interconnection of the central controllers
- processor
- memory
- I/O bus

The high-quality RAM test is a very special feature. It is processed only after an unbuffered POWER ON and in the ERROR-SEARCH mode. The test requires approximately 8 seconds per megabyte of RAM.

Processing self-tests

After an unbuffered POWER ON (for example, POWER ON after plugging in the CPU for the first time or POWER ON without a back-up battery) and in ERROR-SEARCH mode, the CPU executes the complete self-test program. The processing time of the full self-test depends on the configuration of the S7-400H and lasts approximately 30 to 90 sec.

At RUN the operating system divides the self-test into small program sections (test slices) that are processed consecutively over a whole number of cycles. The self-test is organized so that it runs once to the end within 90 minutes. (The 90 minutes are a default value which you can modify by configuring.)

If an error is found as a result of the self-test, the following happens:

- In the event of a hardware error, the CPU enters the cause of the error in the diagnostic buffer and goes to the “Defective” mode.
- In the event of a RAM/PAA comparison error the cause of the error is entered in the diagnostic buffer and the configured system or operating mode is entered.

The master CPU continues to operate at Solo mode.

Influencing the cyclical self-test

With the SFC 90 H_CTRL You can also affect the scope and execution of the cyclical self-test. For example, you can remove and replace individual components of the test. In addition, certain test components can be explicitly called and started for execution.

You will find detailed information on the SFC 90 H_CTRL in the manual on *System Software for S7-300/400, System and Standard Functions*.

Note

With a fail-safe system, the periodic self-tests must not be inhibited and then enabled again.
For more details refer to the manual *S7-400F and S7-400FH Programmable Controllers; Fail-Safe Systems.*

4.2 System Modes of the S7-400H

The system modes of the S7-400H result from the operating modes of the two CPUs. The term “system mode” is used to obtain a simplified expression which identifies the concurrent operating modes of the two CPUs.

Example: Instead of “the master CPU is at RUN and the standby CPU is in the LINK-UP mode” we use “the S7-400H is in the Link-Up system mode”.

Overview of the system modes

The following table shows the system modes that are possible with the S7-400H.

Table 4-1 Overview of the S7-400H System Modes

System Modes of the S7-400H	Operating Modes of the Two CPUs	
	Master	Standby
Stop	STOP	STOP, Power off, Defective
Startup	STARTUP	STOP, Power off, Defective, No synchronization
Solo Mode	RUN	STOP, ERROR-SEARCH, Power off, Defective, No Synchronization
Link-up	RUN	LINK-UP, STARTUP
Update	RUN	UPDATE
Redundant Mode	RUN	RUN
Hold	HOLD	STOP, Power off, Defective

4.3 Operating Modes of the CPUs

Operating modes describe the behavior of the CPUs at any given point of time. Knowledge of the operating modes of the CPUs is useful for programming startup, the test and the error diagnostics.

Operating modes from POWER ON to the Redundant system mode

Generally speaking, the two CPUs enjoy equal rights so that either CPU can be the master or the standby CPU. For reasons of legibility, the illustration presupposes that the master CPU (CPU 0) is energized prior to the standby CPU (CPU 1) being energized.

Figure 4-2 analyzes the operating modes of the two CPUs from POWER ON up to the Redundant system mode. The HOLD mode (refer to Section 4.3.5) and ERROR-SEARCH (refer to Section 4.3.6) are not listed since they are a special case.

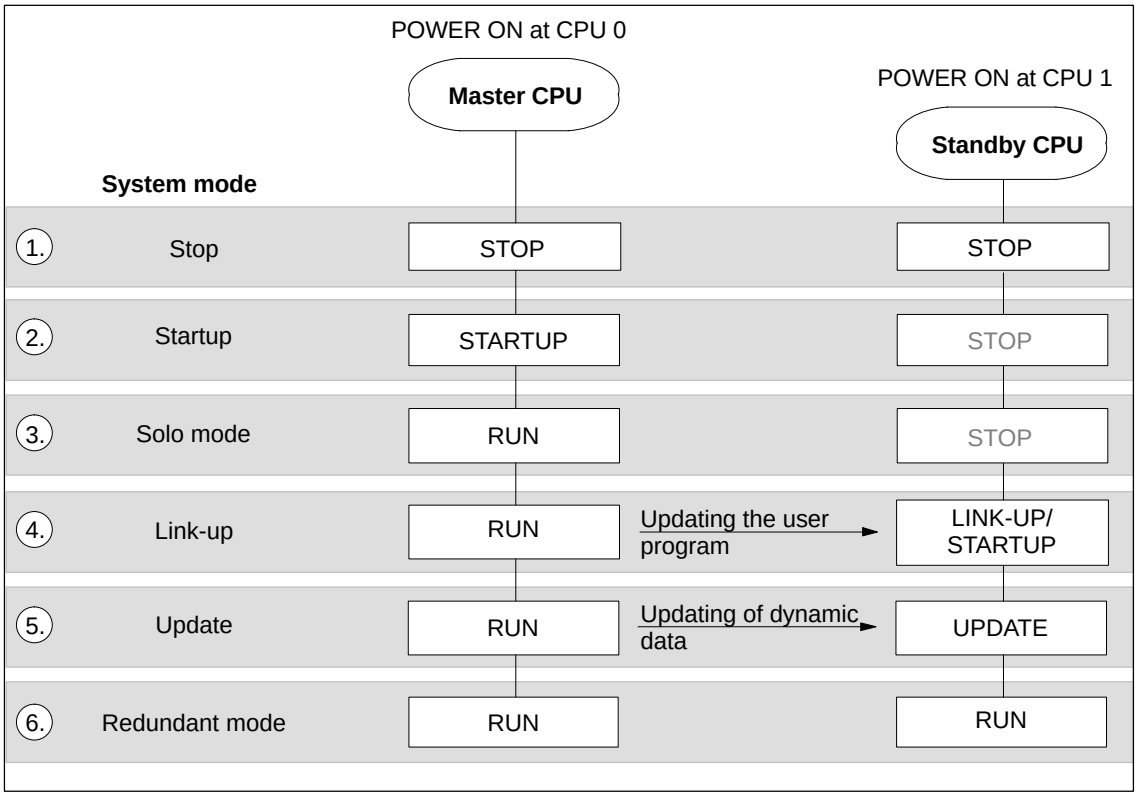


Figure 4-2 System and Operating Modes of the Fault-tolerant System

Explanations relating to Figure 4-2

Table 4-2 Explanations Relating to Figure 4-2 System and Operating Modes of the Fault-tolerant System

Item	Description
1.	Once the power supply has been turned on, the two CPUs (CPU 0 and CPU 1) are in the STOP mode.
2.	CPU 0 goes to the STARTUP mode and processes OB 100 or OB 102, depending on the type of startup. (Refer to Section 4.3.2)
3.	If startup has been successful, the master CPU (CPU 0) changes to Solo mode (the master CPU processes the user program on its own).
4.	If the standby CPU (CPU 1) requests LINK-UP, the master and standby CPUs compare their user programs. If they establish differences, the master CPU updates the user program of the standby CPU. (Refer to Section 4.3.3)
5.	After a successful link-up the update starts (see section 5.2.2). In this instance the master CPU updates the dynamic data of the standby CPU (dynamic data are inputs, outputs, timers, counters, memory markers and data blocks). The contents of the memories of both the CPUs are identical following updating. (Refer to Section 4.3.3)
6.	The master and standby CPUs are at RUN following updating. The two CPUs process the user program in synchronism. Exception: in the case of master/standby switch-over for configuration/program modifications. The Redundant system mode is only possible if both CPUs are of the same release and the same firmware version.

4.3.1 STOP Operating Mode

Except for the additions described below, the CPUs of the S7-400H behave in exactly the same way in STOP mode as the standard CPUs on the S7-400 do.

When both CPUs are in STOP mode and you want to load a configuration, you must make sure you load it into the master CPU. Only then are the system data blocks transferred to the I/O modules.

Memory Reset

Memory Reset always affects only that CPU to whose function it is applied. If you want to reset both CPUs, you must do so first for one and then for the other.

4.3.2 STARTUP Operating Mode

Except for the additions described below, the CPUs of the S7-400H behave in exactly the same way in the STARTUP mode as the standard CPUs on the S7-400 do.

Types of startup

CPU 417-4H distinguishes between a cold restart and a complete restart (warm restart).

Hot restart is not supported by CPU 417-4H.

Startup processing by the master CPU

The startup system mode of an S7-400H is processed entirely by the master CPU; the standby CPU plays no part in startup.

At STARTUP the master CPU compares the existing I/O configuration with the hardware configuration that you created with STEP 7. If there are differences, the master CPU reacts in the same way a standard CPU would on the S7-400.

The master CPU checks and assigns parameters to the

1. switched I/O
2. one-sided, single-channel I/O.

Further information

You will find detailed information on the STARTUP mode in the *Programming with STEP 7* manual.

4.3.3 LINK-UP and UPDATE Operating Modes

Before the fault-tolerant system accepts the Redundant system mode the master CPU checks and updates the memory contents of the standby CPU. (Exception: in the case of link-up and update with subsequent switch-over to CPU with modified configuration).

Checking and updating of the memory contents are performed in two phases which run consecutively and will be referred to in the following as “link-up” and “update”.

During link-up and update the master CPU is always at RUN and the standby CPU is in LINK-UP or UPDATE mode.

When executing a link-up and update a distinction is made between whether the Redundant system mode or a master/standby switch-over is to be achieved (for master/standby switch-over for configuration modifications see Chapter 10). Detailed information on the link-up and update process can be found in section 5.2

4.3.4 RUN Operating Mode

Except for the additions described below, the CPUs of the S7-400H behave in exactly the same way in the RUN mode as the standard CPUs on the S7-400 do.

The user program is executed by at least one of the two CPUs in the following system modes:

- Solo mode
- Link-up, Update
- Redundant mode

Solo mode, Link-up, Update

In the above-named system modes the master CPU is at RUN and executes the user program on its own.

Redundant system mode

The master CPU and the standby CPU are at RUN in the Redundant system mode. The two CPUs execute the user program in synchronism and perform mutual checks.

In the Redundant system mode it is not possible to test the user program with breakpoints.

The Redundant system mode is only possible if both CPUs are of the same release and the same firmware version. It is quit upon the causes of error listed in Table 4-3.

Table 4-3 Causes of Error Leading to the Redundant system mode Being Quit

Cause of Error	Reaction
Failure of one CPU	Refer to Section 9.1.1
Failure of the redundant link (synchronization submodule or fiber-optic cable)	Refer to Section 9.1.5
Error upon comparison of RAM (comparison error)	Refer to Section 4.3.6

Redundant modules

The following rule applies in Redundant system mode:

Redundant modules (for example, the DP slave interface module IM 153-2) must be identical – in other words, they must exhibit the same order number and the same product version and firmware version.

4.3.5 HOLD Operating Mode

With the exception of the additions described below, the S7-400H in HOLD mode behaves in exactly the same way as a regular CPU on the S7-400.

The HOLD mode is a special case. It is only used for test purposes.

When is the HOLD mode possible?

The HOLD mode can be reached only from the STARTUP mode and from RUN submode of Solo mode.

Properties

- While the CPU 417-4H is in the HOLD mode, link-up and update are not possible; the standby CPU remains at STOP and a diagnostic message is issued.
- If the S7-400H is in the Redundant system mode, breakpoints cannot be set.

4.3.6 ERROR-SEARCH Operating mode

During the self-test the master CPU and the standby CPU compare the contents of their memories. If the test discovers different memory contents, a comparison error is reported.

The preset reaction to a comparison error is the ERROR-SEARCH mode (default reaction). The purpose of the search for errors is to detect and localize a faulty CPU.

You can modify the error reaction to a comparison error by means of configuration – for example, the standby CPU is required to go to STOP instead of ERROR-SEARCH in the event of a comparison error.

Error search sequence in Solo mode

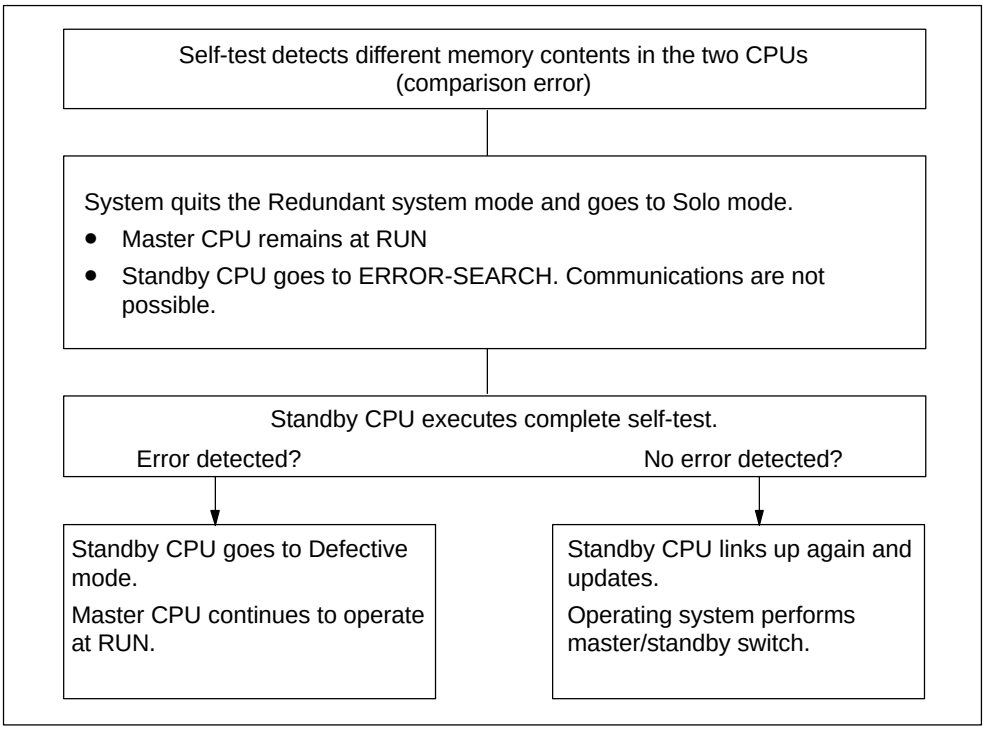


Figure 4-3 Error Search Sequence

If the self-test discovers a comparison error, the S7-400H quits the Redundant system mode and the standby CPU continues working in ERROR SEARCH mode.

During the search for errors the standby CPU executes the entire self-test; the master CPU remains at RUN.

Reaction to recurring comparison error

The reaction to a recurring comparison error depends on whether the error occurs in the subsequent self-test cycle or not until later.

Table 4-4 Reaction to Recurring Comparison Error

Comparison Error Occurs Again ...	Reaction
In the first self-test cycle subsequent to error search	Standby CPU goes to STOP with the diagnostics entry “comparison error”. Master CPU remains at RUN
After two or more self-test cycles subsequent to error search	Standby CPU goes to ERROR-SEARCH. Master CPU remains at RUN

4.4 Time Response

Instruction run times

The run times of the STEP 7 instructions will be found in the instruction list for the S7-400 CPUs.

Executing I/O direct accesses

Please note that every I/O access necessitates synchronization of the two subsystems, thus resulting in a longer scan time.

You should therefore avoid I/O direct accesses in your user program and instead use access via process images (or process image sections – for example, for cyclic interrupts). This produces higher performance since a whole set of values have to be synchronized simultaneously in the case of process images.

Reaction time

You will find detailed information on calculating the reaction times in the *S7-400/M7-400 Programmable Controllers, Module Specifications* Reference Manual.

Note that updating the standby CPU extends the interrupt reaction time (see also section 5.3.1).

The interrupt reaction time depends on the priority class since a graduated delay of the interrupts is performed during updating.

Link-up and Update

5

5.1 Effects of Link-Up and Update

Link-up and update are indicated by the REDF LEDs on the two CPUs. On link-up these LEDs flash with a frequency of 0.5 Hz, and on update with a frequency of 2 Hz.
Link-up and update have various effects on the execution of the user program and the communication functions.

Table 5-1 Properties of Link-Up and Update

Process	Link-Up	Update
Execution of the user program	All priority classes (OBs) are executed.	Execution of the priority classes is delayed by sections. All the requirements are caught up with after the update. You will find the details in the sections which follow.
Deletion, loading, generation and compression of blocks	Blocks cannot be deleted, loaded, generated or compressed. If these actions are being executed, link-up and update will not be possible.	Blocks cannot be deleted, loaded, generated or compressed.
Execution of communication functions, PG operation	Communication functions are being executed.	Execution of the functions is restricted and delayed by sections. All the delayed functions are caught up with after the update. You will find the details in the chapters which follow.
CPU self-test	Will not be performed	Will not be performed
Test and commissioning functions, such as “Monitor and Control Tag”, “Monitor (On/Off)”	No test and commissioning functions are possible. If these actions are being executed, link-up and update will not be possible.	No test and commissioning functions are possible.
Processing of the connections on the master CPU	All the connections remain; no new connections can be made.	All the connections remain; no new connections can be made. Broken connections will only be re-made after the update
Processing of the connections on the standby CPU	All the connections are broken; no new connections can be made.	All the connections are already broken. They are aborted on link-up.

5.2 Functional Sequence of Link-Up and Update

There are two types of link-up and update:

- In a “normal” link-up and update the fault-tolerant system should change from Solo mode to the **Redundant** system mode. The two CPUs then process the same program in synchronously.
- On a link-up and update with **master/standby switch-over** the second CPU with modified components may take over the process control. Either the hardware configuration or the memory configuration or the operating system may be modified.

  In order to return to the Redundant system mode a “normal” link-up and update must be performed subsequently.

How to start link-up and update

Initial situation: Solo mode, i.e. only one of the CPUs of a fault-tolerant system connected via fiber-optic cables is in RUN mode.

Link-up and update to achieve the Redundant system mode can be initiated as follows:

- Change the position of the mode switch on the standby from STOP to RUN or RUN-P.
- POWER ON at the standby (mode switch position RUN or RUN-P), if prior to POWER DOWN the CPU was not in STOP mode.
- Operator control at the PG/ES.

You can only start link-up and update with master/standby switch-over through operator control at the PG/ES.

Note

If link-up and update on the standby CPU is interrupted (for example, POWER DOWN, STOP) inconsistent data may result in a Reset request on this CPU. After the standby is reset link-up and update will be possible again.

Process diagram for link-up and update

The following illustration outlines the functional sequence of link-up and update in general terms. The starting point is with the master in Solo mode. In the illustration CPU 0 is assumed to be the master CPU.

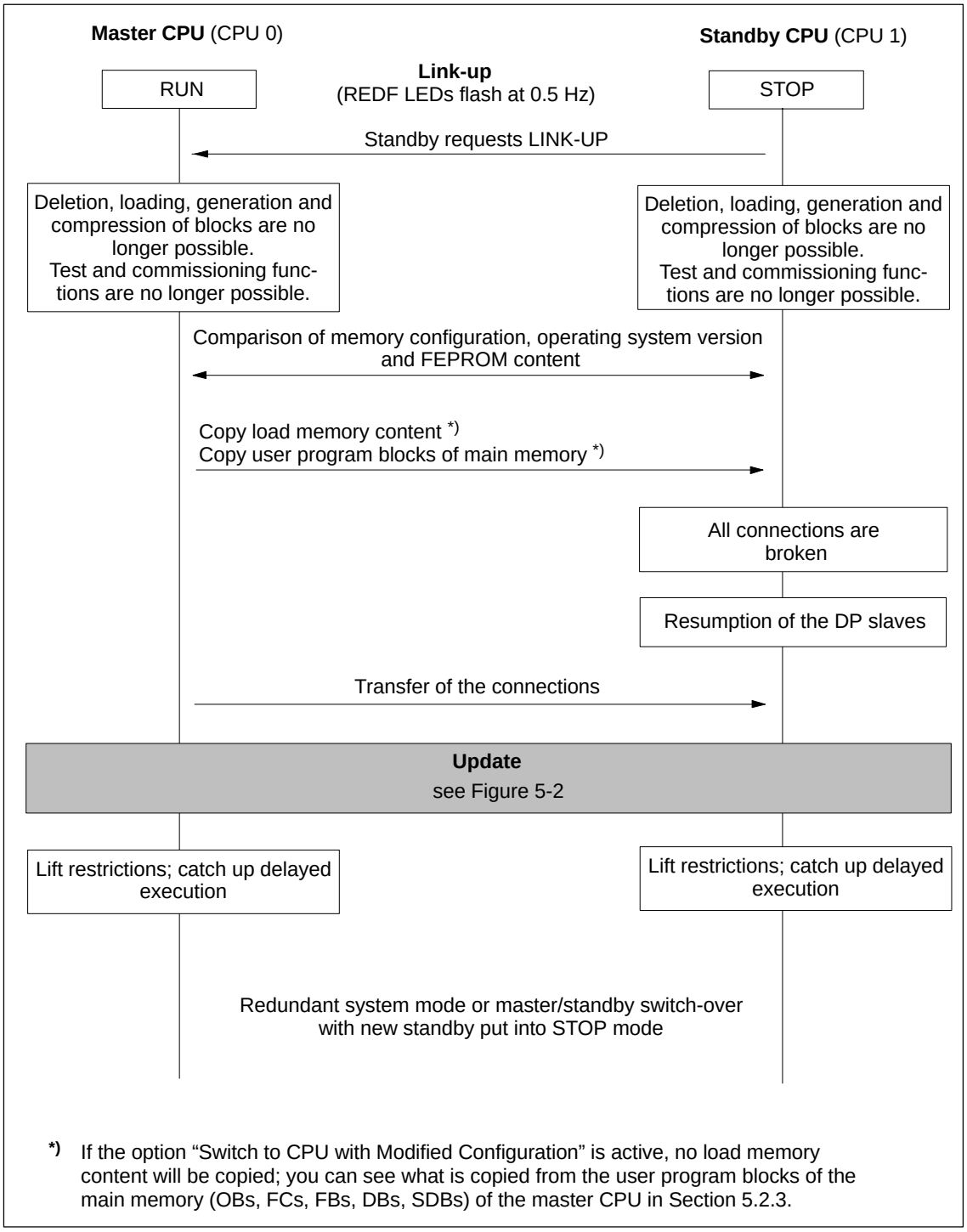


Figure 5-1 Functional Sequence of Link-Up and Update

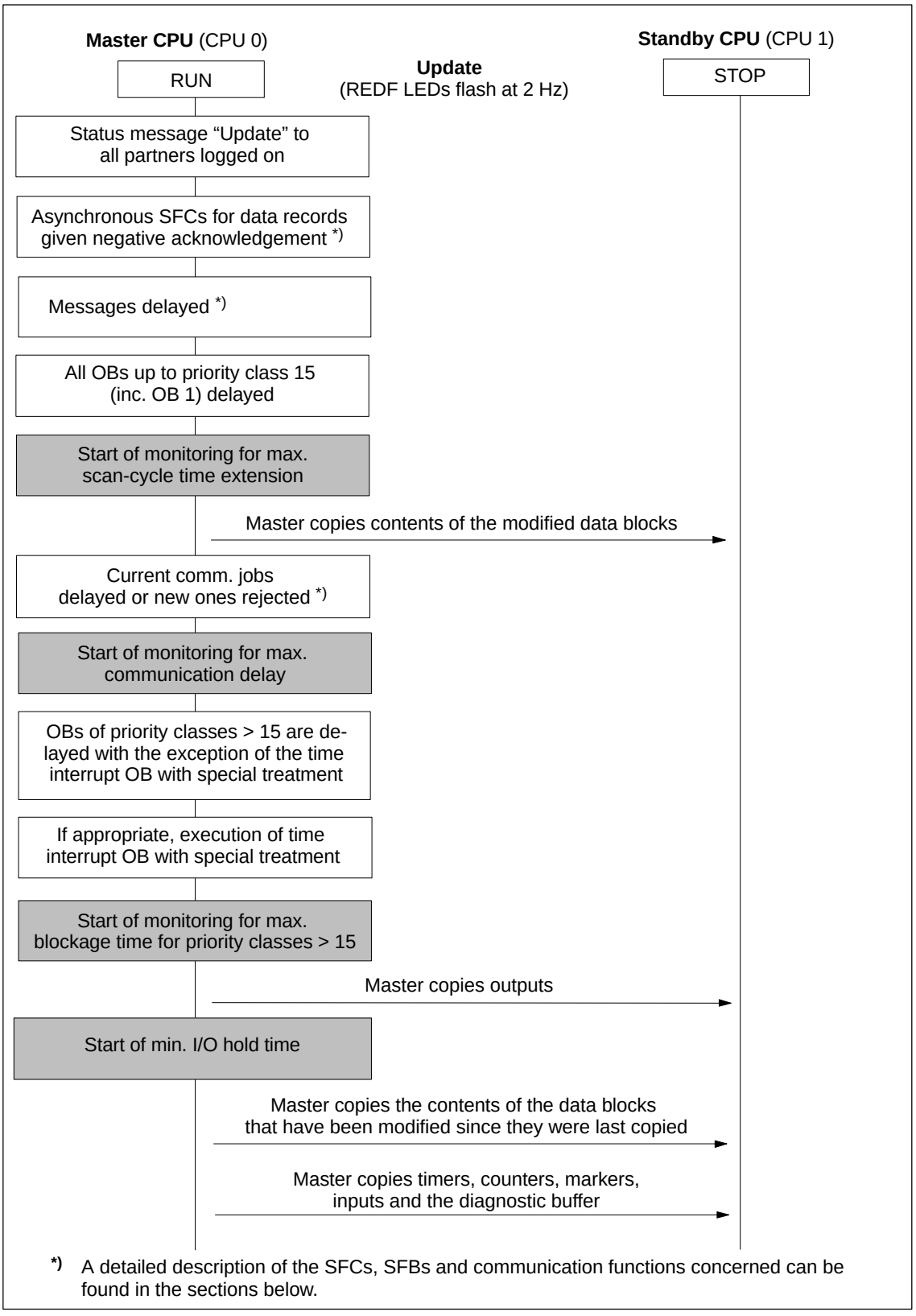


Figure 5-2 Process for update

Minimum signal duration of input signals during the update

During the update, program scanning is stopped for a certain time (we will discuss this subject in greater detail later). So that the change of an input signal can be reliably detected by the CPU even during the update, the following condition must be satisfied:

Minimum signal duration > 2 × time required for I/O update (for DP only)
+ call interval of the priority class
+ processing time for the program of the priority class
+ time for the update
+ processing time for programs of high-priority priority classes

Example:

Minimum signal duration of an input signal that is evaluated in a priority class > 15 (for example, OB 40).

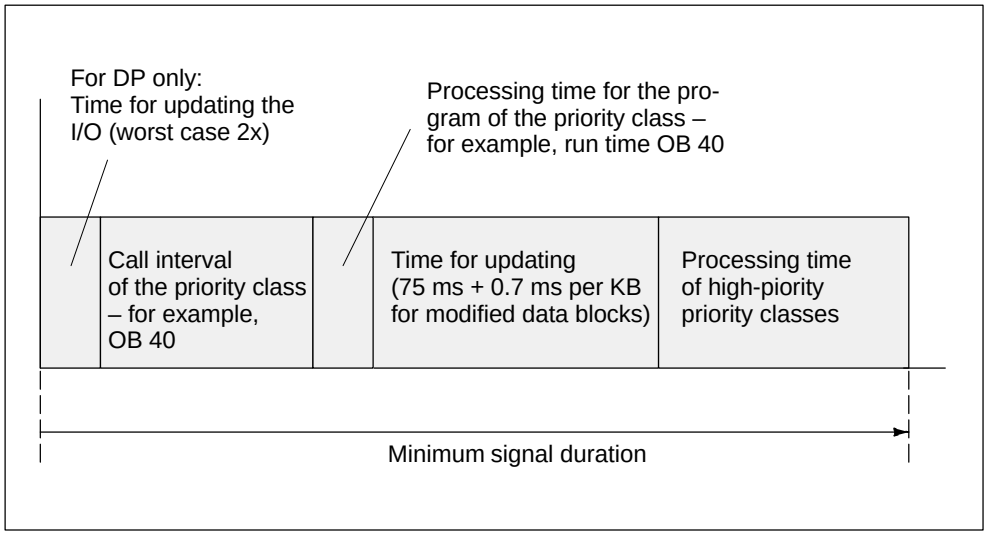


Figure 5-3 Example of Minimum Signal Duration of an Input Signal during Updating

5.2.1 Process of Link-up

In the link-up process a distinction is made between whether the Redundant system mode or a master/standby switch-over is to be achieved.

Link-up to achieve the Redundant system mode

In order to preclude differences in the two subsystems, the master CPU and the standby CPU perform the following comparisons.

The following are checked:

1. the identity of the memory configuration
2. the identity of the operating system version
3. the identity of the content of the load memory (FEPROM card)
4. the identity of the content of the load memory (integrated SRAM and RAM card)

If 1., 2. or 3. are not identical, the standby CPU switches to STOP mode with an error message.

In the case of non-identity of 4. the master CPU copies the user program in the load memory of its RAM to the standby CPU.

The user program in the load memory of the FEPROM is not copied.
It must be identical before link-up.

Link-up with master/standby switch-over

In STEP 7 you can choose one of the following options and thus release the following:

1. Switch to CPU with modified configuration
2. Switch to CPU with extended memory configuration
3. Switch to CPU with modified operating system

Switch to CPU with modified configuration

You may have made the following memory modifications on the standby CPU:

- the hardware configuration
- the type of load memory (e.g. you have replaced a RAM card with a FLASH card); the new load memory may be larger or smaller than the old one.

On link-up no blocks are copied from the master to the standby (the exact circumstances are described in section 5.2.3).

The steps to be performed in the above-mentioned scenarios (modification of hardware configuration, change of type of load memory) are described in Chapter 10.

Note

If you have not changed either the hardware configuration or the type of load memory on the standby CPU a master/standby switch-over is still carried out and the previous master CPU switches to STOP mode.

Switch to CPU with extended memory configuration

You may have made the following memory modifications on the standby CPU:

- enlargement of the main memory and/or
- enlargement of the load memory – in so doing you must have load memory module of the same type, i.e. either RAM cards or FLASH cards; in the case of FLASH cards the content must agree.

On link-up the user program blocks (OBs, FCs, FBs, DBs, SDBs) of the master are copied from the load memory and the main memory to the standby (exception: if the load memory modules are FLASH cards only the blocks from the main memory are copied).

The steps to be performed in the above-mentioned scenarios (enlargement of the main memory, enlargement of the load memory) are described in Chapter 10.

Note

If you have changed the type of load memory or the operating system on the standby CPU this will not switch to RUN mode but instead will relapse into STOP mode with a corresponding diagnostics buffer entry.
If you have enlarged neither the main memory nor the load memory on the standby CPU this will not switch to RUN mode but instead will relapse into STOP mode with a corresponding diagnostics buffer entry.
No master/standby switch-over will take place and the CPU that has been the master CPU up to that point will remain in RUN mode.

Switch to CPU with modified operating system

On the standby CPU you have replaced the operating system with a newer operating system to match the operating system of the master CPU.

On link-up the user program blocks (OBs, FCs, FBs, DBs, SDBs) of the master are copied from the load memory and the main memory to the standby (exception: if the load memory modules are FLASH cards only the blocks from the main memory are copied).

The steps to be performed when replacing the operating system are described in Chapter 10.

Note

If you have not changed the type of load memory or enlarged the main memory or the load memory on the standby CPU this will not switch to RUN mode but instead will relapse into STOP mode with a corresponding diagnostics buffer entry.
If you have not changed the operating system on the standby CPU this will not switch to RUN mode but instead will relapse into STOP mode with a corresponding diagnostics buffer entry.
No master/standby switch-over will take place and the CPU that has been the master CPU up to that point will remain in RUN mode.

5.2.2 Process of Updating

What happens during update?

On update the execution of the communication functions and of the OBs is restricted by section. Similarly, all the dynamic data (content of the data blocks, timers, counters and memory markers) are transferred to the standby CPU.

The update procedure is as follows:

1. All asynchronous SFCs that have recourse to data records of I/O modules (SFC 13, 51, 55 to 59) are given a “negative” acknowledgement until the end of the update:
   - A current job returns BUSY = TRUE. It will be fully executed after the update is complete.
   - A job interrupted during the update will return the value W#16#80C3 (SFCs 13, 55 to 59) or W#16#8085 (SFC 51) once the update is complete. With these return values the jobs should be repeated by the user program.
   - A job that you want to start during the update will be rejected with the return value W#16#80C3 (SFCs 13, 55 to 59) or W#16#8085 (SFC 51). With these return values the jobs should be repeated by the user program once the update is complete.
2. Notification functions are delayed until the end of the update (see list below).
3. The execution of the OB 1 and of all OBs up to and including priority class 15 is delayed.

   With watchdog interrupts, the generation of new OB requests is inhibited so that no new watchdog interrupts are stored and, consequently, no new request errors occur.

   Not until the end of the update is a maximum of one request generated and processed for each watchdog interrupt OB. The time stamp of the watchdog interrupts that are generated after a delay cannot be evaluated.

4. Transfer of all the data block contents that have been modified the since link-up.
5. Communication jobs from which the CPU itself derives jobs for other modules (e.g. I/O) are given a negative acknowledgement (see list below).
6. Initial calls (in other words, calls resulting in manipulation of the work memory, refer also to the *system software for S7-300/400 system and standard functions*) of communication functions receive a negative acknowledgement. All the remaining communication functions are delayed and caught up on once the update is complete.
7. The generation of new OB request for all OBs (in other words, also for those having a priority class > 15) is inhibited so that no new interrupts are stored and, consequently, no request errors occur.

   Not until the end of the update are the queued interrupts requested again and processed. The time stamp of the interrupts that are generated after a delay cannot be evaluated.

   The user program is not executed any more and there are no more I/O updates.
8. Generation of the start event for the watchdog interrupt OB with special handling if its priority class > 15 and execution of this OB, as necessary.

Note

The watchdog interrupt OB with special treatment is of particular significance if it must respond within a particular time to modules or program segments. This is typically the case for fail-safe systems. For more details see the manuals for *S7-400F and S7-400FH Programmable Controllers, Fail-Safe Systems* and *S7-300 Programmable Controllers; Fail-Safe Signal Modules*.

9. Transfer the outputs and the entire data block contents that have been modified. Transfer the timers, counters, memory markers and inputs. Transfer the diagnostic buffer

   During this data synchronization, the clock pulse for watchdog interrupts, delay interrupts and S7 timers is stopped. This results in any synchronism that might have existed between watchdog interrupts and time-of-day interrupts being lost.
10. Lift all restrictions. Delayed interrupts and communication functions will be caught up. All OBs continue to be executed.

    Equidistance from the previous calls can no longer be guaranteed for delayed watchdog interrupt OBs.

Note

Process interrupts and diagnostic interrupts are stored by the I/O. If such interrupts were set by modules of the remote input/output station they will be caught up when the block is lifted. If they were set by modules of the central I/O they can then only all be caught up if the particular interrupt request did not occur again during the block.

If the PG/ES requested a master/standby switch-over then on conclusion of the update the previous standby CPU becomes the master and the previous master CPU switches to STOP mode. Otherwise the two CPUs go to RUN (Redundant mode) and execute the user program in synchronism.

If a master/standby switch-over has been performed then in the next cycle after the update OB 1 has its own identifier (see Reference Manual *System Software for S7-300/400, System and Standard Functions*). For other peculiarities when the configuration is changed see section 5.2.3.

Delayed notification functions

The SFCs, SFBs and operating system services listed trigger messages to all the parameters logged on in each case. These functions are delayed after the start of the update.

- SFC 17 “INTERRUPT_SQ”, SFC 18 “INTERRUPT_S”
- SFC 52 “WR_USMSG”
- SFB 33 “INTERRUPT”, SFB 34 “INTERRUPT_8”, SFB 35 “INTERRUPT_8P”, SFB 36 “NOTIFY”, SFB 37 “AR_SEND”
- Symbol-related messages
- Statuses
- System diagnostics messages

From this time on instructions to block and release events via the SFC 9 “EN_MSG” and the SFC 10 “DIS_MSG” will be rejected with a negative return value.

Communication functions with derived jobs

If a CPU receives one of the jobs listed below it must in turn generate communication jobs from this and send these to other modules. This could be, for example, instructions to read or write parameter data records from/to modules of the remote input/output station. These jobs will be rejected until the update is complete.

- Read/write data records via O & M functions
- Read data records via SSL information
- Block and release messages

- Log on and off for messages
- Acknowledge messages

Note

The last 3 functions are recorded by a WinCC system and automatically repeated when the update is complete.

5.2.3 Switch to CPU with modified configuration

If link-up and update was triggered from STEP 7 using the option “Switch to CPU with modified configuration” the behavior will be different as regards processing of the memory content.

Load memory

Components of the load memory are not copied to the standby CPU by the master CPU.

Main memory

The following components are transferred from the main memory of the master CPU to the standby CPU:

- Contents of all the data blocks having the same interface time stamp in the two load memories and with the attributes “Read Only” and “unlinked” not set.
- Data blocks generated in the master CPU by SFC.

  The DBs generated in the standby CPU by SFC are reset.

  If a data block with the same number is also contained in the load memory of the standby CPU link-up will be interrupted with an entry in the diagnostic buffer.
- Process images, timers, counters and memory markers
- Diagnostic buffer

  If the diagnostic buffer in the standby CPU is configured smaller than in the master CPU only the number of entries for which the standby CPU is configured will be transferred. The most current entries will be selected from the master CPU.

If there is insufficient memory the link-up will be interrupted with an entry in the diagnostic buffer.

If data blocks that contain instances of SFBs of the S7 communication information have been modified then these instances will be returned to the state they were in before first being called.

Note

When switching to a CPU with a modified configuration the load memories of the master and standby may be of different sizes.

5.2.4 Block Link-up and Update

Link-up and update is associated with a scan-cycle time extension. Within this there is a margin of time in which no I/O updating is performed (see section 5.3 “Time Monitoring”). This must be particularly observed if using remote I/O and a master/standby switch-over takes place after the update (i.e. in the event of a configuration modification whilst in operation).

Caution

Only perform link-up and update in non-critical process states.

To specify the start time of the link-up and update yourself, the function SFC 90 “H_CTRL” is available. You will find a detailed description of this SFC in the manual on *System Software for S7-300/400, System and Standard Functions*).

Note

If the process tolerates a longer scan-cycle time extension at any point of time, there is no need to call the 90 “H_CTRL” SFC.

The CPU self-test is not performed during link-up and updating. In the case of a fail-safe system, therefore, make sure that you do not delay the update over too long a period. For more details refer to the manual *S-7-400F and S7-400FH Programmable Controllers; Fail-Safe Systems*.

Example of a time-critical process

A slide block with a 50 mm long cam moves on an axis at a constant velocity v = 10 km/h = 2.78 m/s = 2.78 mm/ms. On the axis is a push button. The push button is thus activated by the cam during a time margin of Δt = 18 ms.

In order for the activation of the push button to be recognized by the CPU the blocking time for priority classes > 15 (see below for definition) must be clearly below 18 ms.

Since in STEP 7 you are able to set the maximum blocking time for priority classes > 15 only to 0 ms or a value of between 100 and 60000 ms, you need to remedy the situation with one of the following measures:

- Move the start of link-up and update to a time at which the process state is not critical. To do this, use SFC 90 “H_CTRL” (see above).
- Use a substantially longer cam and / or clearly reduce the velocity of the slide block before it reaches the push button.

5.3 Time Monitoring

During the update program scanning is stopped for a particular duration. Section 5.3 will be relevant to you if this duration is critical for your process. If so, configure one or more of the monitoring times described below.

During the update the fault-tolerant system will monitor to check that the scan-cycle time extension, the communication delay and the blocking time for priority classes > 15 do not exceed the maximum values configured for them; at the same time it ensures that the minimal I/O hold time configured is maintained.

Note

If you have not specified a value for one or more monitoring times then the associated monitoring will not be activated.

If you have not specified a value for any of the monitoring times then you must take the update into account in the cycle monitoring time. If the update is interrupted in this case, the fault-tolerant system will switch to Solo mode: The CPU that has been the master CPU up to that point will remain in RUN mode and the standby CPU will go to STOP mode.

You took the technological requirements into consideration in the monitoring times configured.

The monitoring times are explained in more detail below.

- Max. scan-cycle time extension
  - Scan-cycle time extension the time margin during the update in which there is no execution of OB 1 (and no execution of any other OBs up to priority class 15). During this margin of time the “normal” cycle time monitoring is ineffective.
  - Max. scan-cycle time extension the maximum permissible scan-cycle time extension that you have configured.
- Max. communication delay
  - Communication delay: the margin of time during the update during which no communication functions are executed. (Note: The existing communication links of the master CPU are maintained.)
  - Maximum communication delay: the maximum permissible communication delay that you have configured.
- Max. blocking time for priority classes > 15
  - Blocking time for priority classes > 15: the margin of time during the update during which no OB (and thus no user program) is executed and no more I/O updates are performed.
  - Max. blocking time for priority classes > 15: the max. permissible blocking time that you have configured for priority classes > 15.

- Minimum I/O hold time:

  This is the period of time between copying of the outputs from the master CPU to the standby CPU and the time of transition to the Redundant system mode or master/standby switch-over (time at which the former master CPU switches to STOP mode and the new master CPU switches to RUN mode). During this time the outputs of both CPUs are activated. This prevents ramping of the I/O even in the event of an update with master/standby switch-over.
  The min. I/O hold time is of particular significance in an update with master/standby switch-over.

The start times of the monitoring timers are shown in Figure 5-2 (underlaid boxes). In each case the times end when the Redundant system mode occurs or on master/standby switch-over (i.e. when the new master switches to RUN mode) at the end of the update.

The times relevant to the update are summarized in the figure below.

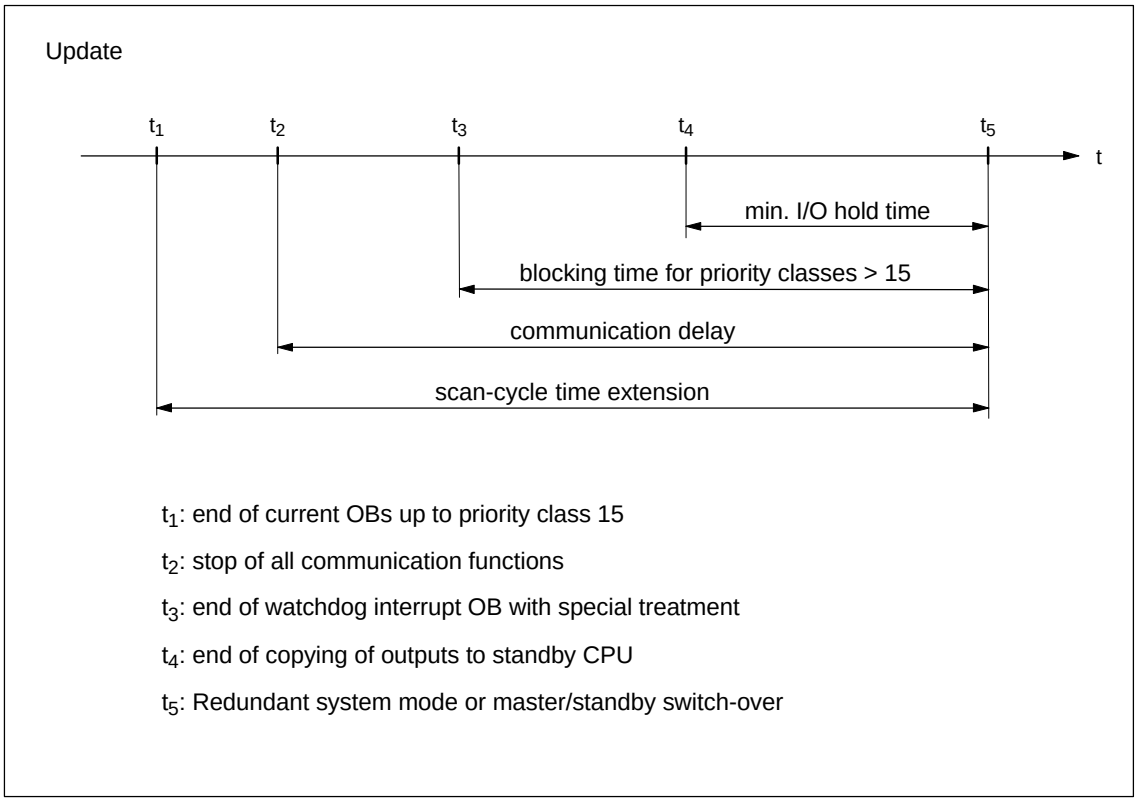


Figure 5-4 Significance of the times relevant during the update

Reaction to time-out

If one of the times monitored exceeds the maximum value configured then the following process is started:

1. Update aborted
2. Fault-tolerant system remains in Solo mode with existing master CPU in RUN mode
3. Reason for aborting entered in the diagnostic buffer
4. OB 72 called (with corresponding start information)

The standby CPU then re-evaluates its system data blocks.
Afterwards – but at least one minute later – there is a new attempt at link-up and update. If unsuccessful after 10 attempts no further attempts will be made. You must then initiate link-up and update again.

Reasons for expiry of the monitoring times may include:

- high interrupt load (i.e. of I/O modules)
- high communication load, so that execution of current functions takes longer
- in the final phase of the update large volumes of data have to be copied to the standby CPU.

5.3.1 Time Behavior

Time behavior during link-up

During link-up, the controller in your system should be influenced as little as possible. Therefore the duration of link-up increases with the rise in the current loading of your programmable logic controller. Link-up duration depends primarily on the

- communication load
- scan cycle time

The following applies to non-loaded programmable logic controllers:

Link-up run time = size of the load and work memories in MB $\times$ 1 s + basic load

The basic load is a few seconds.

When your programmable logic controller is subjected to a high load, the memory-dependent share can rise to 1 minute per MB.

Time behavior during the update

The transfer time during updating depends on the number and overall length of the modified data blocks; it does not depend on the modified volume of data within a block. It is also dependent on the current process
state and on the communication load.

In a simplified view, the max. blocking time to be configured for priority classes > 15 can be seen as a function of the volume of data in the main memory. The volume of code in the main memory is irrelevant.

5.3.2 Determination of the Monitoring Times

Determining with STEP7 or using formulas

There are two ways in which you can determine the monitoring times:

- using the “Calculate...” button on the “H Parameters” tab of the Properties window of the CPU in HWCONFIG (refer to the online Help)
- by using the formulas and steps described below. They are equivalent to the formulas used in STEP7.

Monitoring time accuracy

Note

The monitoring times determined by STEP7 or by using the formulas merely represent a recommendation.

They are based on a fault-tolerant system with two communication peers and an average communication load.

Since your system profile may vary sharply from this assumption, you must take note of the following rules.

- The rise in scan cycle time can increase sharply at a high communication load.
- If you perform changes to your system while it is operating, the rise in scan cycle time can increase appreciably as a result.
- The more program scanning (especially processing of communication blocks) you perform in priority classes > 15, the higher the communication delay and scan cycle time delay might grow.
- You can even undercut the calculated monitoring times in small systems with high performance requirements.

Use of redundant input and output modules

Note

If you have redundant I/O modules and have taken this into account in your program accordingly, you might have to add a premium to the calculated monitoring times, so that surging does not occur at the output modules.

A premium is required only if you operate modules redundantly from the following table.

Table 5-2 Premium for the monitoring times of redundant I/O

Module type	Premium in ms
ET200M: standard output modules	2
ET200M: HART output modules	10
ET200M: fail-safe output modules	50
ET200L-SC with analog output modules	≤ 80
ET200S with analog output modules or technology modules	≤ 20

Perform the following steps:

- Determine the premium form the table. If several module types in the table are used in Redundant mode, take the highest premium.
- Add it to the monitoring times already determined.

Configuration of the monitoring times

When configuring the monitoring times you must note the following dependencies; conformity will be checked by STEP 7:

max. scan-cycle time extension
> max. communication delay
> (max. blocking time for priority classes > 15)
> min. I/O hold time

If, in the case of link-up and update with master/standby switch-over, the CPUs have been configured with different values for a monitoring function then the higher of the two values will be used.

Calculation of the min. I/O hold time (T_{PH})

The following applies to the calculation of the min. I/O hold time:

- with central I/O: T_{PH} = 30 ms
- with remote I/O: $T_{PH} = 3 \times T_{TRmax}$

  where T_{TRmax} = maximum target rotation time of all DP master systems of the H station

With the use of central and remote I/O the resulting minimum I/O hold time is:

T_{PH} = MAX (30 ms, $3 \times T_{TRmax}$)

Figure 5-5 shows the relationship between the min. I/O hold time and the max. blocking time for priority classes > 15.

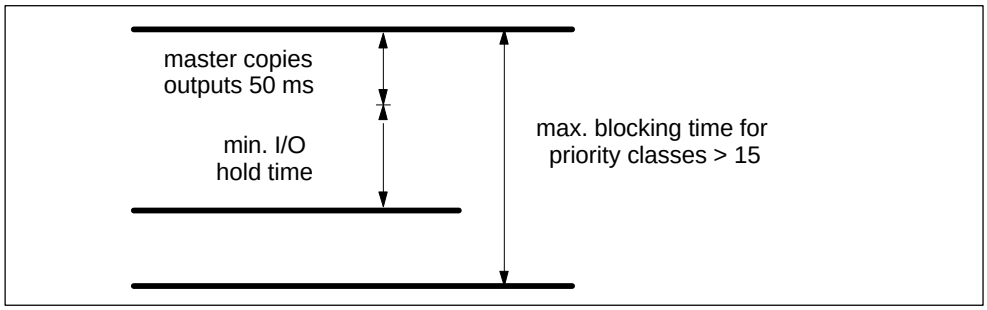


Figure 5-5 Relationship between the min. I/O hold time and the max. blocking time for priority classes > 15.

Note that

50 ms + min. I/O hold time ≤ (max. blocking time of priority classes > 15)

must apply. As a result, a large min. I/O hold time may determine the max. blocking time for priority classes > 15.

Calculation of the max. blocking time for priority classes > 15 (T_{P15})

The max. blocking time for priority classes > 15 is determined by 4 factors:

- As shown in Figure 5-2, at the end of the update all the contents of data blocks that have been modified since the last copy to the standby CPU are transferred to the standby CPU again. **The number and structure of the data blocks** that you describe in the high-priority priority classes determine the duration of this process and thus the max. blocking time for priority classes > 15. A note is given for the remedies specified below.

- In the final phase of the update all the OBs are delayed or blocked. To avoid the max. blocking time for priority classes > 15 being unnecessarily extended as a result of unfortunate symbolic programming, modify the time-critical I/O components in a **selected watchdog interrupt**. This is particularly relevant in the case of fail-safe user programs. You can specify this watchdog interrupt in the configuration; it will then be executed again after the start of the max. blocking time for priority classes > 15, but only if you have assigned it a priority class > 15.
- On link-up and update with master/standby switch-over (see section 5.2.1) then on conclusion of the update the active communication channel in the switched DP slaves must be switched over. This extends the time for which no valid values can be read or output. The duration of this process is determined by your **hardware configuration**.
- The **technological circumstances of your process** give rise to requirements in respect of how long the I/O update can be deferred. This is particularly important in the case of time-monitored processes in fail-safe systems.

Note

Other particular circumstances when using fail-safe modules may be found in the manuals under *S7-400 F and S7-400 FH Programmable Controllers; S7-300 Fail-Safe Systems* and *Programmable Controllers; Fail-Safe Signal Modules*. This applies in particular to module-internal run times in fail-safe modules.

1. For each DP master system determine from the bus parameters in STEP 7
   - T_{TR} for the DP master system
   - DP switch-over time (referred to below as T_{DP_UM})
2. For each DP master system determine from the Technical Data for the switched DP slaves
   - the maximum switch-over time for the active communication channel (referred to below as T_{SLAVE_UM}).
3. From the technological specifications for your system determine
   - the maximum permissible time period for which your I/O modules are not updated (referred to below as T_{PTO}).
4. From your user program determine
   - the scan time of the highest priority or selected (see above) watchdog interrupt (T_{WA})
   - the run time of your program in this watchdog interrupt (T_{PROG})
5. For each DP master system this gives rise to

$$T_{P15}\,(\text{DP master system}) = T_{PTO} - (2 \times T_{TR} + T_{WA} + T_{PROG} + T_{DP\_UM} + T_{SLAVE\_UM}) \quad [1]$$

Note

If T_{P15}(DP master system) < 0 the calculation is to be stopped here. Possible remedies are listed after the following example calculation. Make the appropriate modifications and start the calculation from 1 again.

6. Select the minimum of all the T_{P15} (DP master system) values.
   This time is then known as T_{P15_HW}.
7. Determine the share of the maximum blocking time for I/O classes > 15 determined by the minimum I/O hold time (T_{P15_OD}):

   T_{P15_OD} = 50 ms + min. I/O hold time [2
   ]

Note

If $T_{P15\_OD} > T_{P15\_HW}$ the calculation is to be stopped here. Possible remedies are listed after the following example calculation. Make the appropriate modifications and start the calculation from 1 again.

8. From section 5.3.4 determine the share of the maximum blocking time for priority classes > 15 determined by the user program (T_{P15_AWP}).

Note

If $T_{P15\_AWP} > T_{P15\_HW}$ the calculation is to be stopped here. Possible remedies are listed after the following example calculation. Make the appropriate modifications and start the calculation from 1 again.

9. The recommended value for the max. blocking time for priority classes > 15 now results from:

   $T_{P15} = MAX\ (T_{P15\_AWP}, T_{P15\_OD})$ [3]

Example of calculation of T_{P15}

In the following the maximum permissible period of time on update during which the operating system performs no program scanning and no I/O updates is determined for a given system configuration.

There are two DP master systems: DP master system_1 is “connected” to the CPU via the MPI/DP interface of the CPU and DP master system_2 via an external DP master interface module.

1. From the bus parameters in STEP 7:

   T_{TR_1} = 25 ms

   T_{TR_2} = 30 ms

   $T_{DP\_UM\_1}$ = 100 ms

   $T_{DP\_UM\_2}$ = 80 ms

2. From the Technical Data for the DP slaves used:

   $T_{SLAVE\_UM\_1}$ = 30 ms

   $T_{SLAVE\_UM\_2}$ = 50 ms

3. From the technological specifications for your system:

   T_{PTO_1} = 1250 ms

   T_{PTO_2} = 1200 ms

4. From the user program:

   T_{WA} = 300 ms

   T_{PROG} = 50 ms

5. From formula [1]:

   T_{P15} (DP master system_1)

   = 1250 ms – (2 $\times$ 25 ms + 300 ms + 50 ms + 100 ms + 30 ms) = 720 ms

   T_{P15} (DP master system_2)

   = 1200 ms – (2 $\times$ 30 ms + 300 ms + 50 ms + 80 ms + 50 ms) = 660 ms

   Check: if $T_{P15} > 0$, continue with

6. T_{P15_HW} = MIN (720 ms, 660 ms) = 660 ms

7. From formula [2]:

   T_{P15_OD} = 50 ms + T_{PH} = 50 ms + 90 ms = 140 ms

   Check: if T_{P15_OD} = 140 ms < T_{P15_HW} = 660 ms, continue with

8. From section 5.3.4 for 170 Kbytes user program data:

   T_{P15_AWP} = 194 ms

   Check: if T_{P15_AWP} = 194 ms < T_{P15_HW} = 660 ms, continue with

9. Formula [3] now provides the recommended max. blocking time for priority classes > 15:

   T_{P15} = MAX (194 ms, 140 ms)

   T_{P15} = 194 ms

   If you enter 194 ms for the maximum blocking time for priority class > 15 in STEP 7 it is ensured that in a change of signal during the update is always recognized for signal durations of 1250 ms or 1200 ms.

Remedies for if it is not possible to calculate T_{P15}

If no recommendation results from the calculation of the max. blocking time for priority classes > 15 you can remedy this by various measures:

- Reduce the watchdog interrupt cycle of the watchdog interrupt configured.
- For particularly high T_{TR} times divide the slaves into several DP master systems.
- Increase the transmission rate on DP master systems affected.
- If the DP slaves have very different switch-over times and thus (generally) great variations in T_{PTO}, divided these slaves into several DP master systems.
- If little load due to interrupts or parameter assignment is to be expected in the individual DP master systems you can also reduce the T_{TR} times determined by approx. 20–30%. This will increase the risk of a station failure occurring in the remote I/O.
- The time T_{P15_AWP} indicates a guide value; this depends on you program structure. You can reduce it by, for example, placing data that is modified frequently in different DBs to data that is modified less often. Reducing the time T_{P15_AWP} without taking the measures stated increases the risk of the update being aborted due to expiry of the monitoring times.

Calculation of the max. communication delay

We recommend the following formula:

Maximum comms delay = 4 × (maximum blocking time for priority classes > 15)

The final time is determined by the process state and the communication load on your system. This has to be taken to mean both the absolute load and also the load in relation to the size of your user program. You may have to correct the time if necessary.

Calculation of the max. scan-cycle time extension

We recommend the following formula:

Maximum cycle time prolongation = 10 × (maximum blocking time for priority classes > 15)

The final time is determined by the process state and the communication load on your system. This has to be taken to mean both the absolute load and also the load in relation to the size of your user program. You may have to correct the time if necessary.

5.3.3 Influences on the Time Behavior

The period during which no I/O updates take place is primarily determined by the following influencing factors:

- number and size of data blocks modified during the update
- number of instances of SFBs in the S7 communication information and SFBs for generating block-related messages
- modifications to the System while in Operation
- settings via dynamic volume frameworks
- expansion of remote I/O (with falling transmission rate and increasing number of slaves the time required for I/O updates increases.)

In the least favorable cases this period is extended by the following amounts:

- maximum watchdog interrupt cycle used
- duration of all watchdog interrupt OBs
- duration of high-priority interrupt OBs running up until delay of the interrupts

Deliberate delaying of the update

Delay the update via SFC 90 “H_CTRL” and do not release it again until a state of lesser communication or interrupt load occurs.

Caution

Delaying the update will increase the time during which the fault-tolerant system is in Solo mode.

5.3.4 Performance Values for Link-up and Update

User program share T_{P15_AWP} of the max. blocking time for priority classes > 15

The user program share T_{P15_AWP} of the max. blocking time for priority classes > 15 can be calculated using the following formula:

T_{P15_AWP} in ms = 0.7 $\times$ size of the DBs in the work memory in Kbyte + 75

The following table provides the resulting times for some typical values for the main memory data.

Table 5-3 Typical values for the user program share T_{P15_AWP} of the max. blocking time for priority classes > 15

Main memory data	T_{P15_AWP}
500 Kbyte	430 ms
1 Mbyte	800 ms
2 Mbyte	1.51 s
5 Mbyte	3.66 s
10 Mbyte	7.24 s

The following assumptions were made for this formula:

- 80% of the data blocks are modified prior to delaying the interrupts of priority classes > 15.
  This value must be determined more accurately for fail-safe systems in particular in order to avoid a time-out of the driver blocks (see Section 5.3.2).
- Approximately 100 ms of update time are allowed for each megabyte of work memory assigned by data blocks for currently running or held-back communication functions.
  Depending on the communication load of your programmable logic controller, you have to add to or deduct from the setting of T_{P15_AWP}.

5.4 Special Features during Link-up and Update

Requirement of input signals during the update

During the update the process signals read in previously are retained and are not updated. Modification of a process signal during the update will only be recognized by the CPU if the modified signal state remains at the end of the update.

Pulses (signal change “0 → 1 → 0” or “1 → 0 → 1”) occurring during the update will not be recognized by the CPU.

Make sure, therefore, that the time between two signal changes (pulse duration) is always greater than the time required for the update.

Communication links and functions

In contrast to earlier firmware versions, links to the master CPU are no longer broken. However, associated communication jobs are not executed during the update. They are stored and caught up on when one of the following cases occurs:

- the update is complete and the system is in Redundant mode
- the update and master/standby switch-over are complete, the system is in Solo mode
- the update was aborted (e.g. due to a time-out) and the system is back in Solo mode.

It is not possible for the communication blocks to make an initial call during the update.

Reset instruction on aborting link-up

If the link-up is aborted whilst the contents of the load memory are being copied from the master CPU to the standby CPU then the standby CPU will given a Reset instruction. This is signaled by an entry in the diagnostic buffer with the event ID W#16#6523.

Using I/O on the S7-400H

6

This chapter provides an overview of the different I/O configurations on the S7-400H programmable logic controller and its availability. Further, it provides information on configuration and programming of the selected I/O installation.

For the S7-400H you can use virtually any of the input/output modules featured in the SIMATIC S7 system range. This applies to the input/output modules of the S7-400 standard system and to the PROFIBUS-DP components. The function modules (FMs) and communication processors (CPs) that can be used in the S7-400H will be found in Appendix E.

6.1 Introduction

I/O configuration types

In addition to the power supplies and central processing units, which are always redundant, there are the following configuration types for the I/O, which are supported by the operating system:

- Single-channel, one-sided configuration with normal availability
- Single-channel, switched configuration with enhanced availability

A two-channel redundant configuration is similarly possible. However, you have to implement the high degree of availability in the user program (refer to Section 6.4).

Addressing

No matter whether you are using a single-channel, one-sided or switched I/O, you always specify the same address for the I/O.

Limits of I/O configuration

If there are insufficient slots in the central controllers, you can add up to 20 expansion units to the configuration of the S7-400H.

You can assign even-numbered mounting racks only to central controller 0, whereas odd-numbered mounting racks can be assigned only to central controller 1.

For using distributed I/O you can connect up to 12 DP master systems in each of the subsystems (2 DP master systems to the integrated interfaces of the CPU and 10 more via external DP master systems).

You can operate up to 32 slaves on the integrated MPI/DP interface. You can connect up to 125 distributed I/O devices to the integrated DP master interface and the external DP master systems.

6.2 Using a Single-Channel, One-Sided I/O

What is a single-channel, one-sided I/O?

With the single-channel, one-sided configuration single input/output modules are present (single-channel). The input/output modules are located in just one of the subsystems and are only addressed by that subsystem.

A single-channel, one-sided I/O configuration is possible in

- central controllers and expansion units
- distributed I/Os

The single-channel, one-sided I/O configuration is to be recommended for individual input/output channels for which normal availability of the I/O is sufficient.

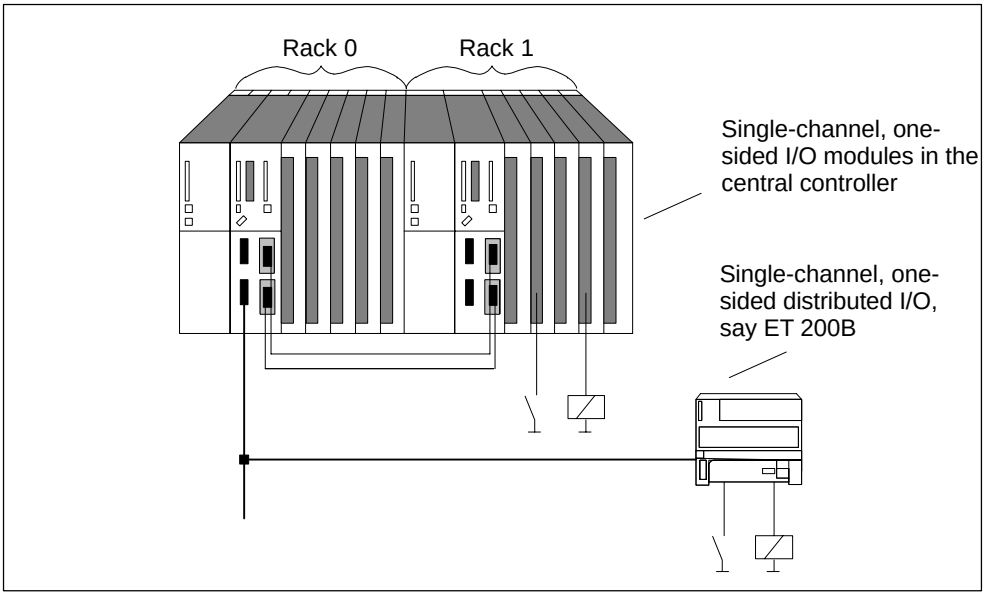


Figure 6-1 Single-Channel, One-Sided I/O Configuration

Single-channel, one-sided I/Os and user program

Information read in on one side – for example, from digital inputs – is transferred automatically to the second subsystem via the synchronization link in the Redundant system mode.

After the information has been transferred, both subsystems have the data from the single-channel, one-sided I/O and evaluate them in the two identical user programs that are present. It is therefore not decisive whether the I/O is connected to the master CPU or to the standby CPU as far as processing of the information in the Redundant system mode is concerned.

In Solo mode, the one-sided I/O assigned to the cooperating subsystem cannot be accessed. This has to be taken into account in your programming as follows: you have to assign functions to the single-channel, one-sided I/O that can only be performed conditionally. In this way you make sure that certain functions for I/O accesses are triggered only in the Redundant system mode and in Solo mode of the subsystem concerned.

Note

The user program has to update the process image for single-channel, one-sided output modules in Solo mode too (e.g. direct accesses). If using subprocess imagers the user program must update the subprocess imager in OB 72 (redundancy return) accordingly (SFC 27 UPDAT_PO). If it did not, old values would initially be read out to the single-channel, one-sided output modules of the standby CPU following transition to the Redundant system mode.

Failure of the single-channel, one-sided I/O

In the event of a malfunction the S7-400H with a single-channel, one-sided I/O behaves like a standard S7-400 system, in other words:

- When the I/O fails, the defective I/O is no longer available.
- When a subsystem fails, the entire process I/O of that subsystem is not available any more.

6.3 Using Single-Channel, Switched I/O

What is a single-channel, switched I/O?

With the single-channel, switched configuration single input/output modules are present (single-channel).

In Redundant mode they may be addressed by both subsystems.

In Solo mode, the master subsystem can always address **all switched I/O** (as opposed to one-way I/O). The non-master subsystem cannot address switched I/O.

The single-channel, switched I/O configuration is possible with the ET 200M distributed I/O device having an active backplane bus and a redundant PROFIBUS-DP slave interface module IM 153-2 or IM 153-2FO (permitted IM 153-2: 6ES7 153-2AA02-0XB0 version 6 or later; permitted IM 153-2FO: 6ES7 153-2AB01-0XB0 version 5 or later). Each subsystem of the S7-400H is connected to one of the two DP slave interfaces of the ET 200M (via a DP master interface).

In addition, DP/PA coupler IM 157 allows a redundant connection to PROFIBUS PQ (permitted IM 157: 6ES7 157-0AA81-0XA0 version 1 or later and firmware version 3.1.0).

The single-channel, switched I/O configuration is recommended for devices that tolerate the failure of individual modules within the ET 200M.

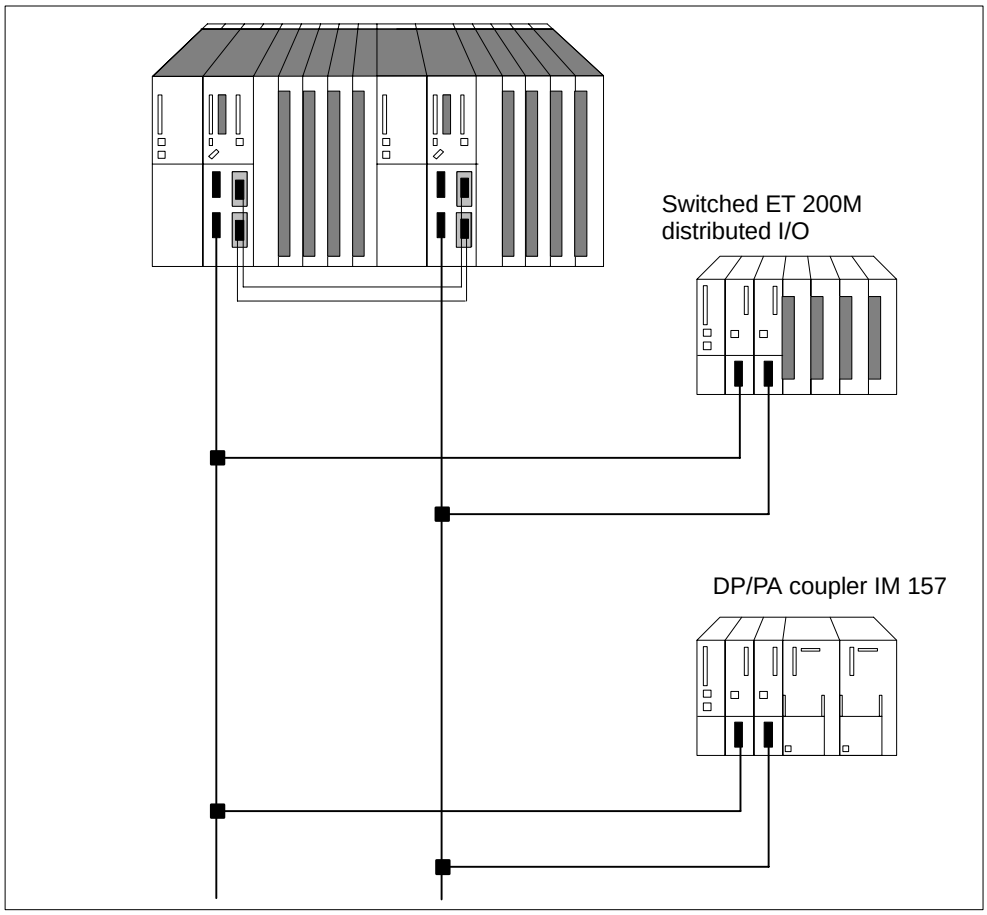


Figure 6-2 Single-Channel, Switched ET 200M Distributed I/O

Rule

When you use a single-channel, switched I/O, the configuration must always be symmetrical, in other words:

- the CPU 417-4 H and other DP masters must be located in both subsystems in identical slots – for example, in slot 4 on both subsystems – or
- the DP masters must be connected on both subsystems to the same integrated interface – for example, to the PROFIBUS-DP interfaces of the two CPUs 417-4 H.

Single-channel, switched I/O and user program

In Redundant mode, in principle each subsystem may access single-channel switched I/O. The information is automatically transferred over the synchronization link and compared. An identical value is available to the two subsystems at all times owing to the synchronized access.

The S7-400H only ever uses one of the interfaces at any given time. The active interface is indicated by the ACT LED on the corresponding IM 153-2 or IM 157.

The path via the currently active interface (IM 153-2 or IM 157) is described as the **active channel** and the path via the other interface as the **passive channel**. The DP cycle always runs via both channels. However, only the input and output values of the active channel are processed in the user program or output to the I/O. The same applies to asynchronous activities such as interrupt processing and the exchange of data records.

Failure of the single-channel, switched I/O

In the event of a malfunction the S7-400H with a single-channel, switched I/O behaves as follows:

- When the I/O fails, the defective I/O is no longer available.
- In certain failure situations, e.g. failure of a subsystem, a DP master system or a DP slave IM153-2 or IM 157 interface module (refer to Chapter 8), the single-channel, switched I/O continues to be available to the process.
  This is achieved by switch-over between the active and slave channel. This switch-over takes place separately for each DP station. On a failure a distinction is made between
  - failures affecting one station only (failure of the DP slave interface module of the currently active channel)
  - failures affecting all the stations of a DP master system.
    These include unplugging the DP master interface module, shutdown of the DP master system (e.g. on RUN-STOP transition on a CP 443-5) and a short circuit in the cable loom of a DP master system.

The following applies to each station affected by a failure: if both DP slave interface modules are currently functional and the active channel fails, the previous slave channel automatically becomes the active channel. A redundancy loss is reported to the user program via the start of OB 70 (event W#16#73A3).

Once the fault has been remedied the redundancy is restored. This again results in the start of OB 70 (event W#16#72A3). In this instance there is no switch-over between the active and slave channel.

If one channel has already failed and the remaining (active) channel also fails, then there is a complete station failure. This results in the start of OB 86 (event W#16#39C4).

Note

If the DP master interface module can recognize failure of the complete DP master system (e.g. in the case of a short circuit), only this event is reported (“Master system failure coming” W#16#39C3). The operating system then no longer reports individual station failures. This allows the switch-over process between the active and slave channel to be accelerated.

Clearance of routing connections upon switchover of the active channel

When the active channel is switched over, all the routing connections to modules and devices located downstream of the IM 153-2(FO) or IM 157 are cleared. You have to re-establish these connections in the user program.

Duration of switch-over of the active channel

The maximum switch-over time is

DP error recognition time + DP switch-over time + switch-over time of DP slave interface module

You can determine the first two addends from the bus parameters of your DP master system in STEP 7. You define the last addend from the manuals of the DP slave interface modules concerned (*Distributed I/O ET 200M* and *DP/PA bus connection*).

Note

The above calculation also includes the processing time in OB 70 or OB 86. Note that the processing for a DP station must take **no more than 1 ms**. If more extensive processing is required, disconnect this from the direct processing of the OBs mentioned.

Note that a change of signal can only be detected by the CPU if the signal duration is greater than the specified switch-over time.
With a switch-over of the complete DP master system, the switch-over time of the slowest component applies to all DP components. The DP/PA link normally defines the transfer time and the associated minimum signal duration. We therefore recommend you to connect DP/PA links to a separate DP master system.

Switch-over of the active channel on link-up and update

On link-up and update with master/standby switch-over (see section 5.2.1) the active and slave channels are switched over in all the stations of the switched I/O. No OBs are started in this instance.

No pulses during switch-over of the active channel

To prevent temporary failure of the I/O or output of substitute values during switch-over between the active and slave channel the DP stations of the switched I/O maintain their outputs until switch-over is complete and the new active channel has taken over processing.

In order that total failures of a DP station occurring during switch-over are also recognized the switch-over process is monitored by both the individual DP stations and by the DP master system.

No interrupts or data records are lost as a result of a switch-over. Automatic repetition takes place if necessary.

System configuration and design

You should sort switched I/O with different transfer times into separate looms. This simplifies calculation of the monitoring times, among other things.

6.4 Connecting a Redundant I/O

What is a redundant I/O?

A redundant I/O consists of input/output modules, of which several are present. If you wish to use a redundant I/O, you can implement it at user level.

Configurations

The following configurations having a redundant I/O are possible (Figure 6-3):

1. Redundant system with one-way central and/or distributed I/O.

   One I/O module is inserted for this purpose in the subsystems of CPU 0 and one in the subsystem of CPU 1.

2. Redundant configuration with a switched I/O.

   Two I/O modules are inserted into two ET 200M distributed I/O devices with an active backplane bus.

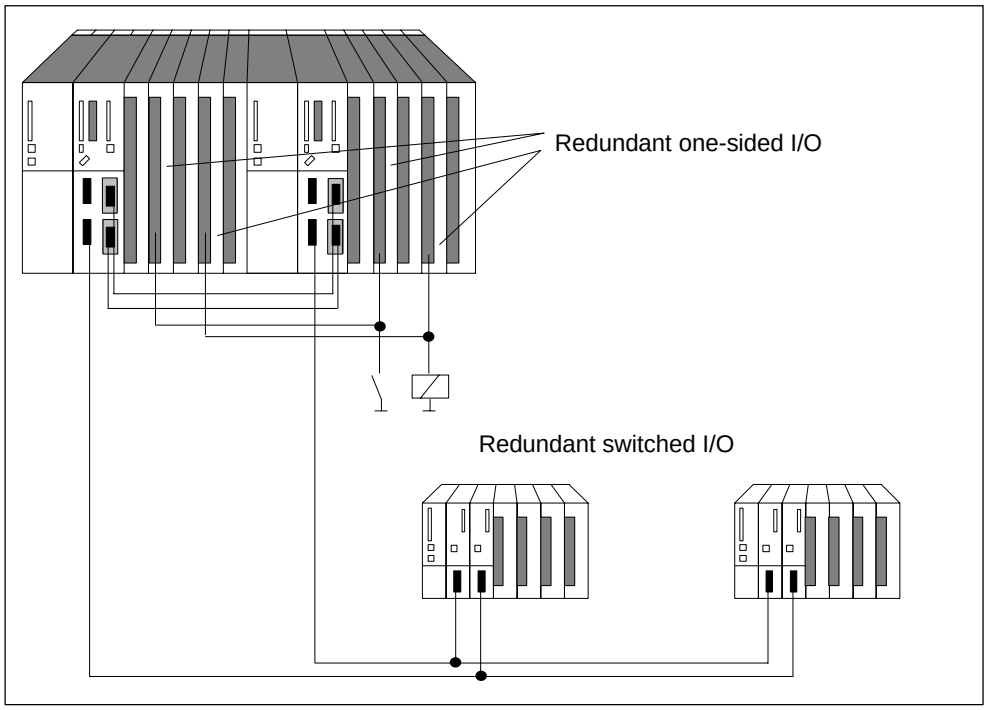


Figure 6-3 Redundant One-Sided and Switched I/Os

Note

When using redundant I/O, a premium might have to be added to the calculated monitoring times; refer to Section 5.3.2.

Hardware installation and configuration of the redundant I/O

If you wish to use a redundant I/O, we would recommend you the following strategy:

1. Use the I/O in the following manner:
   - with a one-sided configuration, one I/O module in each subsystem
   - with a switched configuration, two I/O modules in two distributed ET 200M I/O devices.
2. Wire the I/O in such a way that it can be addressed by both subsystems.
3. Configure the I/O modules for different logical addresses.

Note

We do not recommend configuration of the output modules you use to the same logical addresses as the input modules; if it is done nevertheless, you have to query the type (input or output) of the defective group in OB 122, in addition to the logical address.

The user program must update the process image for redundant one-way output modules in Solo mode too (e.g. direct accesses). If using subprocess imagers the user program must update the subprocess imager in OB 72 (redundancy return) accordingly (SFC 27 UPDAT_PO). If it did not, old values would initially be read out to the single-channel, one-sided output modules of the standby CPU following transition to the Redundant system mode.

Redundant I/O in the user program

The following example program shows the use of two redundant digital input modules:

- module A in rack 0 with logical base address 8 and
- module B in rack 1 with logical base address 12.

One of the two modules is read directly in OB1. It is assumed for the following, without limiting general applicability, that it is module A (the value of variable BGA is TRUE). If no error occurred, processing continues with the value read.

If an I/O access error has occurred, module B will be read by direct access (“second attempt” in OB1). If no error has occurred, processing continues with the value read by module B. However, if an error has similarly occurred in this instance, both modules are currently defective, and work continues with a substitute value.

The example program is based on the fact that following an access error to module A, module B is always processed first in OB1 after the former has been replaced. Module A is not processed first again in OB1 until an access error to module B occurs.

Note

Variables BGA and PZF_BIT must also be valid outside OB1 and OB122. The variable VERSUCH2, on the other hand, is used only in OB1.

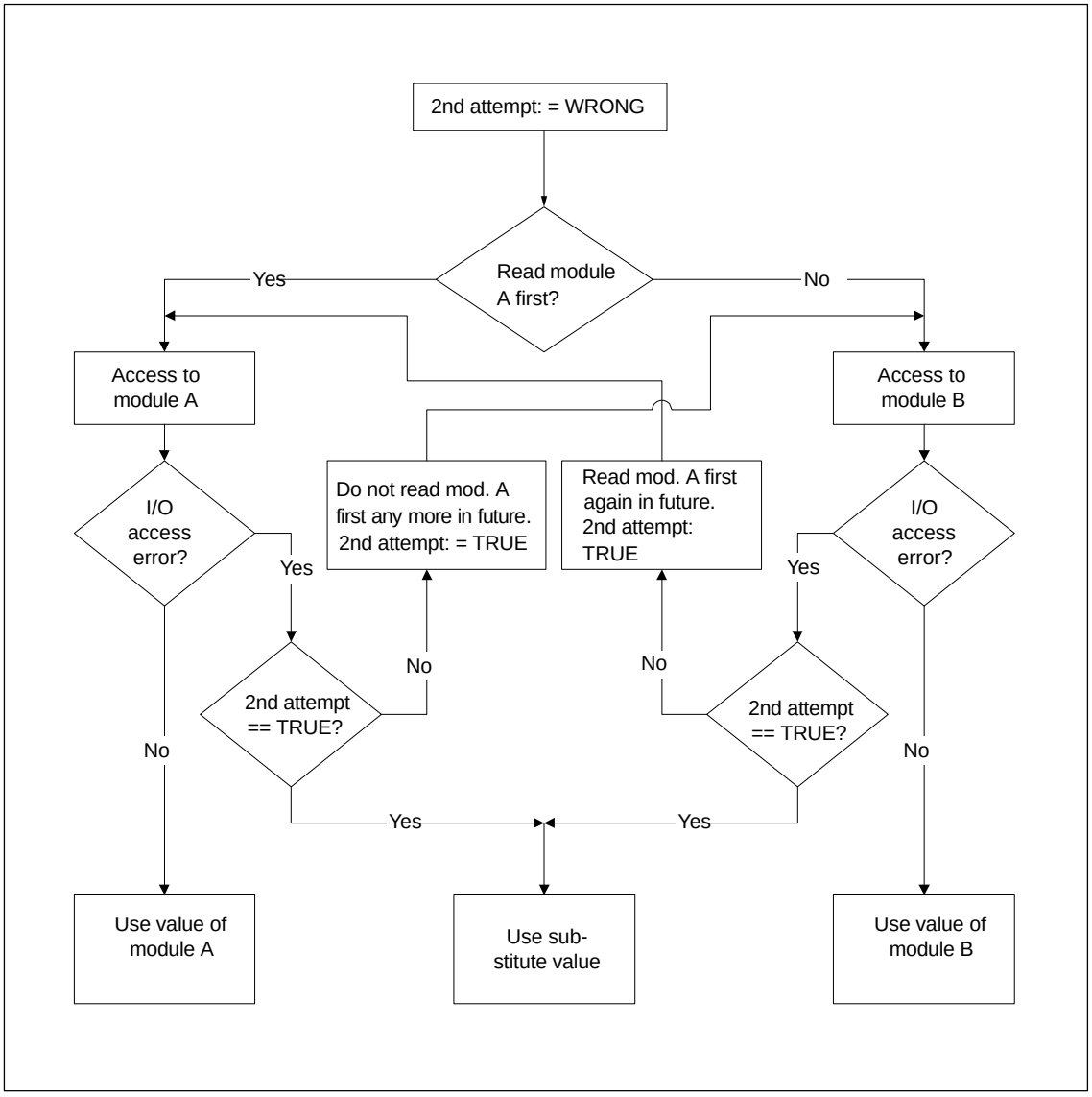


Figure 6-4 Flowchart for OB1

Example of STL

The requisite sections of the user program (OB1, OB 122) are listed below.

Table 6-1 OB 1

STL		Explanation
	NOP 0;	
	SET;	
	R VERSUCH2;	//Initialization
	A BGA;	//Read module A first?
	JCN WBGB;	//If No, continue with module B
WBGA:	SET;	
	R PZF_BIT;	//Delete PZF bit
	L PED 8;	//Read CPU 0
	U PZF_BIT;	//Was PZF detected in OB 122?
	JCN PZOK;	//If No, process access OK
	U VERSUCH2;	//Was this access the second attempt?
	JC WBG0;	//If Yes, use substitute value
	SET;	
	R BGA;	//Do not read module A first any more //in future
	S VERSUCH2;	
WBGB:	SET;	
	R PZF_BIT;	//Delete PZF-Bit
	L PED 12;	//Read CPU 1
	U PZF_BIT;	//Was PZF detected in OB 122?
	JCN PZOK;	//If No, process access OK
	U VERSUCH2;	//Was this access the second attempt?
	JC WBG0;	//If Yes, use substitute value
	SET;	
	S BGA;	//Read module A first again in future
	S VERSUCH2;	
	JU WBGA;	
WBG0:	L ERSATZ;	//Substitute value
PZOK:		//The value to be used is in Accumulator1

Table 6-2 OB 122

STL		Explanation
		// Does module A cause PZF?
	L OB122_MEM_ADDR;	//Logical base address affected
	L W#16#8;	
	== I;	Module A?
	JCN M01;	//If No, continue with M01
		//PZF upon access to module A
	SET;	
	= PZF_BIT;	//Set PZF bit
	JU CONT;	
		// Does module B cause PZF?
M01:	NOP 0;	
	L OB122_MEM_ADDR;	//Logical baseaddress affected
	L W#16#C;	
	== I;	module B?
	JCN CONT;	//If No, continue with CONT
		//PZF upon access to module B
	SET;	
	= PZF_BIT;	//Set PZF bit
CONT:	NOP 0;	

Communications

7

In this chapter you will find an introduction to communications with fault-tolerant systems and their specific characteristics.

You will learn the basic concepts, the bus systems you can use for fault-tolerant communications and the types of connection.

You will learn how communications take place via fault-tolerant connections and standard connections, and how to configure and program them.

- You will find examples of communications via **fault-tolerant S7 connections** and will learn about their advantages.
- By way of a comparison, you will learn in this respect how communications take place over **S7 connections** and also how you can communicate in Redundant mode with S7 connections.

7.1 Fundamentals and Basic Concepts

Overview

Fault-tolerant controllers make it possible for controllers, including their I/O, to feature redundancy. With growing demands on the availability of an overall system it is necessary to raise the fault tolerance of communications – in other words, communications have to be configured so that they are also redundant.

You will find below an overview of the fundamentals and basic concepts which you ought to know with regard to using fault-tolerant communications.

Redundant communication system

The availability of the communication system can be enhanced by redundancy of the media, duplication of subcomponents, or duplication of all bus components.

Monitoring and synchronization mechanisms ensure that should one component fail, communications in routine operation are continued by means of standby components.

A redundant communications system is a requirement for the configuration of fault-tolerant S7 connections.

Fault-tolerant communications

Fault-tolerant communications means the use of SFBs in S7 communications via fault-tolerant S7 connections.

Fault-tolerant S7 connections are only possible with the use of redundant communications systems.

Redundant nodes

Redundant nodes represent the fault tolerance of communications between two fault-tolerant systems. A system having multi-channel components is represented by redundant nodes. The independence of a redundant node is given when the failure of a component within the node does not result in reliability constraints in other nodes.

Connection (S7 connection)

A connection is the logical assignment of two communication peers to implement a communication service. Every connection has two endpoints containing the information required for addressing the communication peer and other attributes for establishing the connection.

An S7 connection is the communication connection between two standard CPUs or from one standard CPU to a CPU in a fault-tolerant system.

In contrast to a fault-tolerant S7 connection, which contains at least two partial connections, an S7 connection actually consists of just one connection Communications are terminated should this one connection fail.

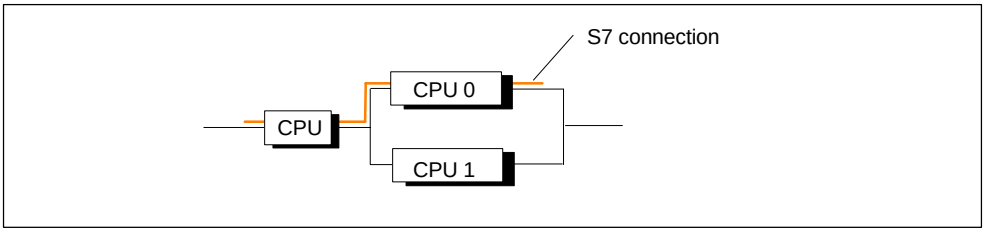


Figure 7-1 Example of an S7 Connection

Note

Generally speaking, “connection” in this manual means the “configured S7 connection”. For other types of connection please refer to the manuals *SIMATIC NET NCM S7 for PROFIBUS* and *SIMATIC NET NCM S7 for Industrial Ethernet*.

Fault-tolerant S7 connections

The requirement for higher availability by means of communication components – for example, CPs and buses – necessitates redundant communication connections between the systems involved.

Unlike the S7 connection, a fault-tolerant S7 connection consists of at least two lower-level partial connections. From the point of view of the user program, the configuration and the connection diagnostics, the fault-tolerant S7 connection with its subordinate partial connections is represented by exactly one ID (like a standard S7 connection). Depending on the configuration set, it can consist of up to four partial connections, of which two are always made (active) so as to maintain communications in the event of an error. The number of partial connections depends on possible alternative paths (refer to Figure 7-2) and is determined automatically.

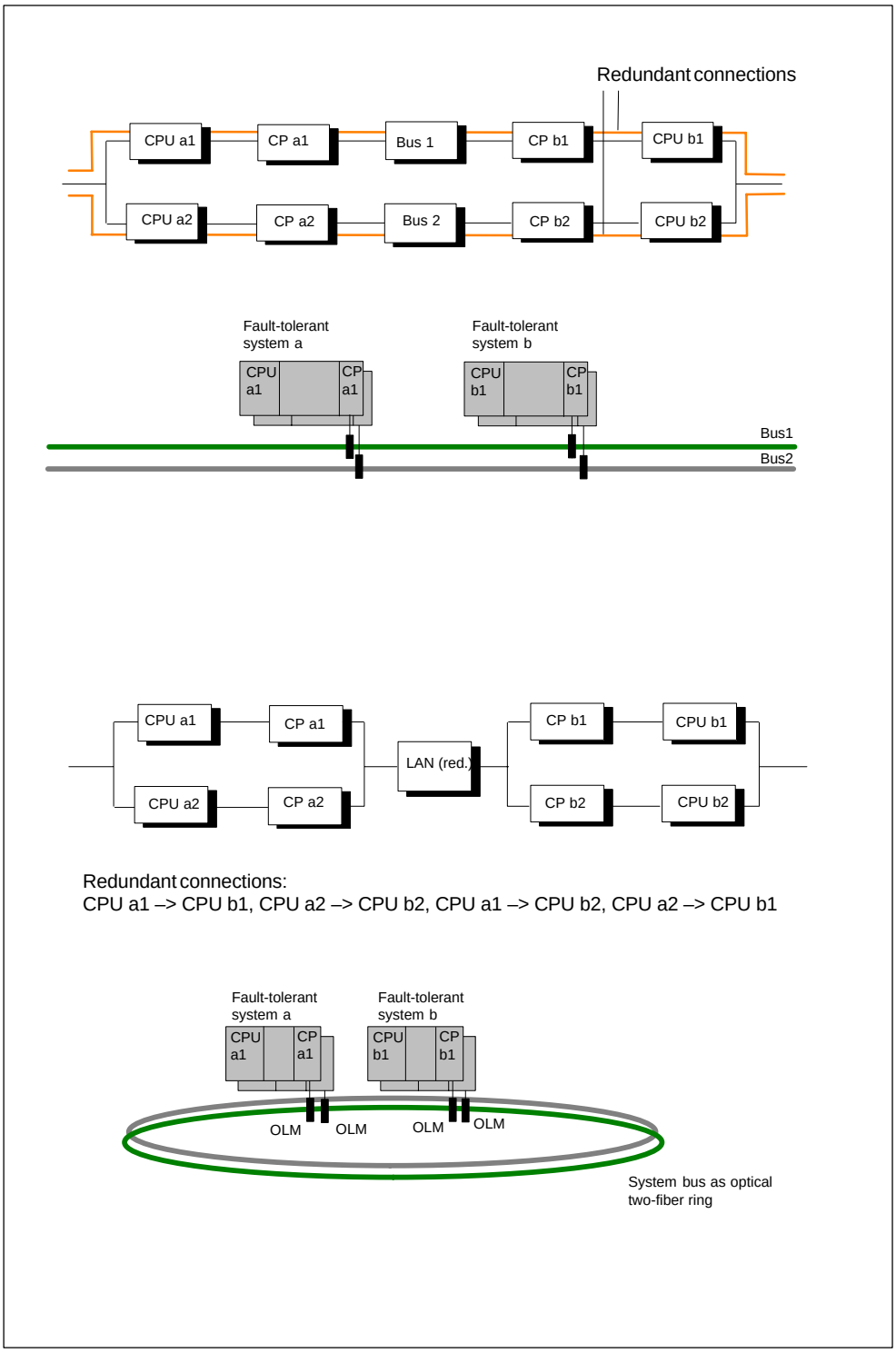


Figure 7-2 Example of the Number of Resulting Partial Connections Being Dependent on the Configuration

Should the active partial connection fail, a previously established second partial connection assumes responsibility for communications.

Resource requirements of fault-tolerant S7 connections

CPU 417-4H allows the operation of 64 fault-tolerant S7 connections. On the CP each partial connection requires a connection resource.

7.2 Suitable Networks

The choice of physical transfer medium depends on the desired expansion, the fault tolerance aimed at and the transmission rate. The following bus systems are used for communications with fault-tolerant systems:

- Industrial Ethernet (fiber-optic cable, triaxial or twisted pair copper cable)
- PROFIBUS (fiber-optic cable or copper cable)

You will find further information on suitable networks in the manuals "*Communication with SIMATIC*", *"Industrial Twisted Pair Networks"* and *"PROFIBUS Networks*".

7.2.1 Industrial Ethernet

The industrial Ethernet is a communications network for cells in baseband transmission technology complying with IEEE 802.3, with the CSMA/CD access procedure.

An industrial Ethernet network can be configured as a redundant medium with either electrical or optical components. A very wide range of electrical and optical network components is available for the industrial Ethernet.

Electrical network

The electrical network can be configured as a classical bus structure having triaxial cabling for the transmission medium.

An addition and an alternative to conventional bus cabling for connections to terminals are electrical link modules (ELMs) or industrial twisted pairs (ITPs). With them star networks complying with IEEE 802.3 can be configured.

Optical network

The optical industrial Ethernet network (transmission medium: fiber-optic cable) can be configured as a line-type, ring or star network. The configuration is accomplished for a transmission rate of 10 Mbps with optical link modules (OLMs) and/or star hubs for the fast Ethernet of 100 Mbps with optical switching modules (OSMs) and optical redundancy manager (ORM).

7.2.2 PROFIBUS

PROFIBUS is a communications network for cells and fields in accordance with PROFIBUS Standard EN 50 170, Volume 2, with the hybrid token bus and master slave access procedure. Networking takes place over two-wire cables or fiber-optic cables.

The PROFIBUS bus system can be used as a redundant medium with electrical or optical components. The number of linked stations should not be greater than 30. The industrial Ethernet bus system is recommended for larger systems.

The transmission rate can be adjusted in steps from 9.6 kbps to 12 Mbps.

Electrical network

The transmission medium of the electrical network is a shielded, twisted pair.

The RS 485 interface operates with voltage differences. It is therefore less sensitive to interference than a voltage or current interface. In the case of PROFIBUS the nodes are connected through a bus terminal or a bus connector to the bus (up to 32 nodes per segment). The different segments are interconnected by means of repeaters.

The maximum segment size depends on the transmission rate.

Apart from RS-485 transmission technology, there is PROFIBUS PA to IEC 1158 for process automation. PROFIBUS PA transmission technology aims at the intrinsically safe Exi area and thus operates with a synchronous, low-energy transmission method. Up to ten nodes can be operated in the intrinsically safe Exi area on a single PROFIBUS PA segment, provided that the overall power requirement never exceeds 100 mA. In the non-intrinsically safe area as many as 30 nodes can operated on one PROFIBUS PA segment. The transmission rate used is then 31.25 kbps.

Optical network

The optical PROFIBUS network uses fiber-optic cables as a transmission medium.

The fiber-optic cable variant is insensitive to electromagnetic interference, is lightning-proof, does not require electrical equipotential bonding and is suitable for long distances (glass fiber-optic cable).

The maximum segment length does not depend on the transmission rate (except redundant optical rings). Optical rings can be configured as one- or two-fiber rings (enhanced network availability).

Configuration of the fiber-optic cable networks is accomplished by means of optical link modules (OLMs). A network can be configured with OLMs as a line-type, ring or star network.

7.3 Supported Communication Services

The following services can be used:

- S7 communications over fault-tolerant S7 connections via PROFIBUS and Industrial Ethernet
- S7 communications over S7 connections via MPI, PROFIBUS and Industrial Ethernet
- standard communications (FMS, for example) via PROFIBUS
- S5-compatible communications (SEND and RECEIVE blocks, for example) via PROFIBUS and Industrial Ethernet

The following are not supported:

- basic communications
- global data communications

7.4 Communications via Fault-Tolerant S7 Connections

Availability of communicating systems

Fault-tolerant communications add additional, redundant communication components, such as CPs and LAN cables, to the overall SIMATIC system. To illustrate the actual availability of communicating systems when using an optical or electrical network, a description is given below of the possibilities for communication redundancy.

Requirements

A requirement for the configuration of fault-tolerant connections with STEP 7 is a configured hardware installation.

The hardware configurations of the two subsystems integrated in a fault-tolerant system **must** be identical. This is especially true of the slots.

Depending on the network being used, the following CPs can be used for fault-tolerant communications:

- Industrial Ethernet:
  S7: CP 443-1
- PROFIBUS:
  S7: CP 443-5 Basic, CP 443-5 Extended (not configured as DP master system)

In order to be able to use fault-tolerant S7 connections between a fault-tolerant system and a PC, the “S7-REDCONNECT” software package is required on the PC. Please refer to the Product Information Leaflet on “S7-REDCONNECT” to learn more about the CPs you can use at the PC end.

Configuration

The availability of the system, including communications, is set during configuration. Please refer to the STEP 7 documentation to find out how to configure connections.

Only S7 communication is used for fault-tolerant S7 connections. To do this, select in the “New Connection” dialog box the “S7 Connection Fault-Tolerant ” as the type.

The number of required redundant connections is determined by STEP 7 as a function of the redundant nodes. If the network architecture allows them, up to four redundant connections are generated. Higher redundancy cannot be achieved by using more CPs.

In the “Properties - Connection” dialog box you can modify specific properties of a fault-tolerant connection, should you require to do so. If you are using more than one CP, you can also switch the connections in this dialog box. This may be practical since, by default, all connections are routed initially through the first CP. If all the connections have been assigned there, the other connections are routed through the second CP etc.

Programming

Fault-tolerant communications can be used on the CPU 417-4H and are accomplished via S7 communication.

This is possible solely within an S7 project.

The programming of fault-tolerant communications with STEP 7 is accomplished by means of communication system function blocks. With these blocks, data can be transmitted over subnets (Industrial Ethernet, PROFIBUS). The standard communication SFBs integrated in the operating system offer you the option of acknowledged data transmission. Not only data transmission is possible; other communication functions for controlling and monitoring the communication peer can also be used.

User programs written for standard communications can be run for fault-tolerant communications as well, without being modified. Cable and connection redundancy has no effect on the user program.

Note

You will find tips on programming communications in the S7 standard documentation – for example, *Programming with STEP 7*.

The communication functions START and STOP act on exactly one CPU or on all CPUs of the fault-tolerant system (for more details refer to the Reference Manual *System software for S7-300/400, System and standard functions*).

7.4.1 Communications between Fault-Tolerant Systems

Availability

The simplest method of enhancing the availability of interconnected systems is to use a redundant system bus configured with an optical two-fiber ring or a duplicated electrical bus system. In this case the connected nodes may consist of simple standard components.

An increase in availability can best be obtained with an optical two-fiber ring. Should the two-fiber optic cable rupture, communications continue to exist between the systems involved. The systems then communicate as if they were connected to a bus system (line). A ring system contains two redundant components as a matter of principle and therefore automatically forms a 1-out-of-2 redundant node. An optical network can also be configured in line-type or star topology. There is no cable redundancy with line-type topology, however.

Should one electrical cable segment fail, communications between the systems involved similarly continue to exist (1-out-of-2 redundancy).

The following examples illustrate the differences between the two versions.

Note

The number of connection resources required on the CPs depends on the network you are using.

If you are using an optical two-fiber ring (refer to Figure 7-3), two connection resources are required per CP. In contrast to this, only one connection resource is required per CP if a duplicated electrical network (refer to Figure 7-4) is being used.

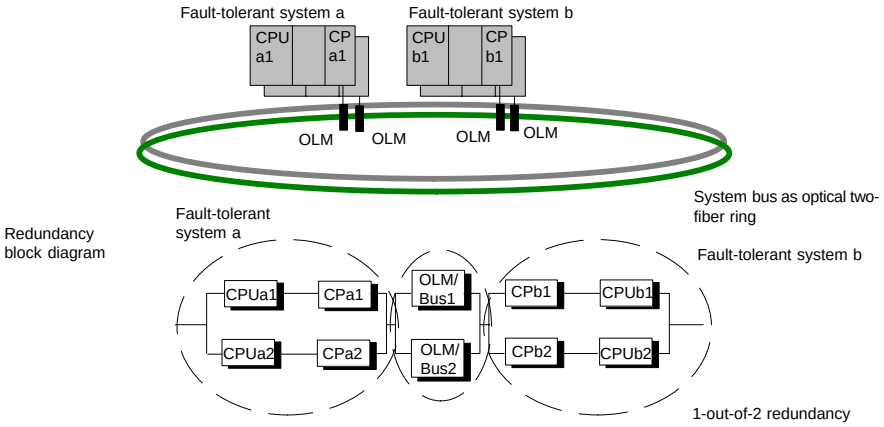


Figure 7-3 Example of Redundancy with Fault-Tolerant System and Redundant Ring

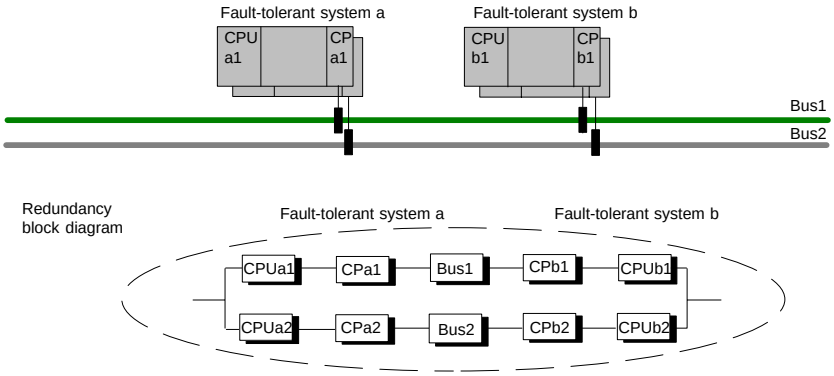


Figure 7-4 Example of Redundancy with Fault-Tolerant System and Redundant Bus System

Failure behavior

Only a double error within a fault-tolerant system (e.g. CPUa1 and CPa2 in a system) in the case of a two-fiber ring leads to total failure of communications between the redundant systems concerned (refer to Figure 7-3).

Should a double error occur on a redundant electrical bus system – for example, CPUa1 and CPb2 – this results in total failure of communications between the systems involved (refer to Figure 7-4).

7.4.2 Communications between Fault-Tolerant Systems and a Fault-Tolerant CPU

Availability

Availability can be enhanced by using a redundant system bus and by using a fault-tolerant CPU on a standard system.

If the communication peer is a CPU 417-4 H, fault-tolerant connections can be configured again here, as opposed to a CPU 416, for example.

Note

Fault-tolerant connections occupy two connection resources on CP b1 for the redundant connections. One connection resource each is assigned to CP a1 and CP a2.

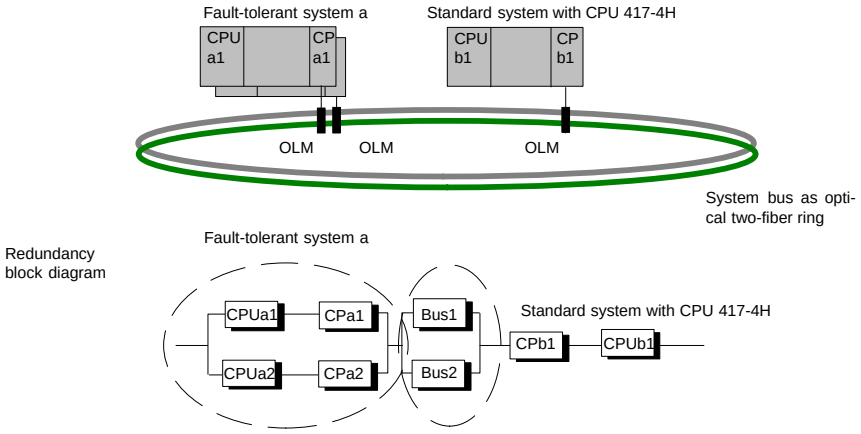


Figure 7-5 Example of Redundancy with Fault-Tolerant System and CPU 417-4H

Failure behavior

Double errors in the fault-tolerant system (i.e. CPUa1 and CPa2) and single errors in the standard system (CPUb1) result in a total failure of communications between the redundant systems involved (refer to Figure 7-5).

7.4.3 Communications between Fault-Tolerant Systems and PCs

Availability

When fault-tolerant systems are connected to a PC, the availability of the overall system concentrates not only on the PCs (OS) and their data management but also on data acquisition on the programmable logic controllers.

PCs are not fault-tolerant on account of their hardware and software characteristics. They can be arranged in a redundant manner in a system, however. The availability of this kind of PC (OS) system and its data management is ensured by means of suitable software such as WinCC Redundancy.

Communications take place via fault-tolerant connections.

The “S7-REDCONNECT” software package is a requirement for fault-tolerant communications on a PC. It makes it possible to connect a PC to an optical network having a CP or to a redundant bus system having two CPs.

Configuring connections

No additional configuration of fault-tolerant communications is required at the PC end. Connection configuration is taken care of by the STEP 7 project in the form of an XDB file at the PC end.

You can find out how to use STEP 7 fault-tolerant S7 communications to integrate a PC in your OS system in the WinCC documentation.

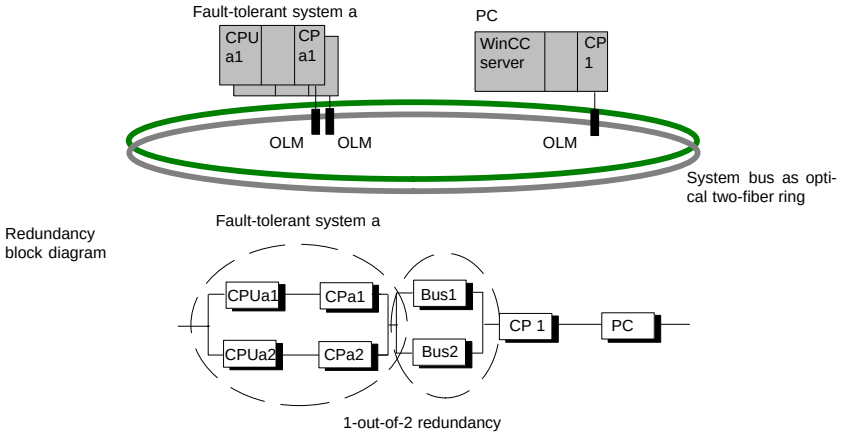


Figure 7-6 Example of Redundancy with Fault-Tolerant System and Redundant Bus System

Failure behavior

Double errors in the fault-tolerant system (i.e. CPUa1 and CPa2) and failure of the PC will result in total failure of communications between the systems involved (refer to Figure 7-6).

7.5 Communications via S7 Connections

Communications with standard systems

Fault-tolerant communications are not possible between fault-tolerant and standard systems. The following examples illustrate the actual availability of the communicating systems.

Configuration

Standard connections are configured with STEP 7.

Programming

If standard communications are used on a fault-tolerant system, all the communication functions apart from “global data communications” can be used for them.

The standard communication SFBs are used for programming communications with STEP 7.

Note

The communication functions START and STOP act on exactly one CPU or on all CPUs of the fault-tolerant system (for more details refer to the Reference Manual *System software for S7-300/400, System and standard functions*).

7.5.1 Communications via S7 Connections – One-Sided Mode

Availability

Availability is similarly enhanced by using a redundant system bus for communications from a fault-tolerant system to a standard system.

If the system bus is configured as an optical two-fiber ring, communications between the systems involved continue to exist in the event of the two-fiber optic cable rupturing. The systems then communicate as if they were connected to a bus system (line).

When fault-tolerant systems and standard systems are interconnected, the availability of communications cannot be enhanced by means of a twin electrical bus system. In order to be in a position to use the second bus system as a redundant bus, you have to use a second S7 connection, which has to be managed accordingly in the user program (refer to Figure 7-7).

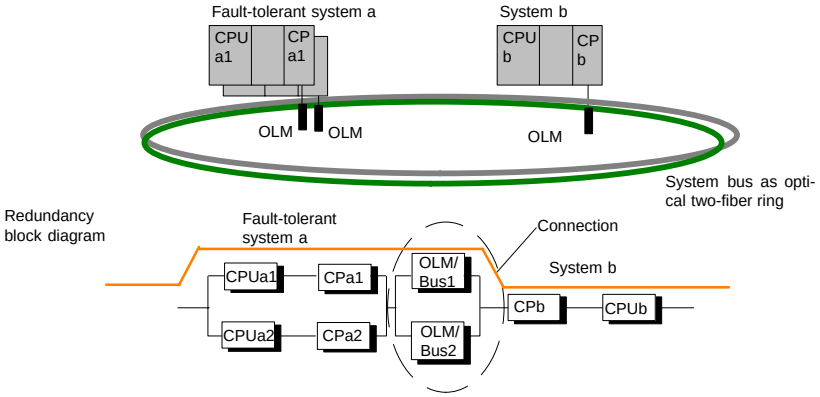


Figure 7-7 Example of Interconnected Standard and Fault-Tolerant Systems on a Redundant Ring

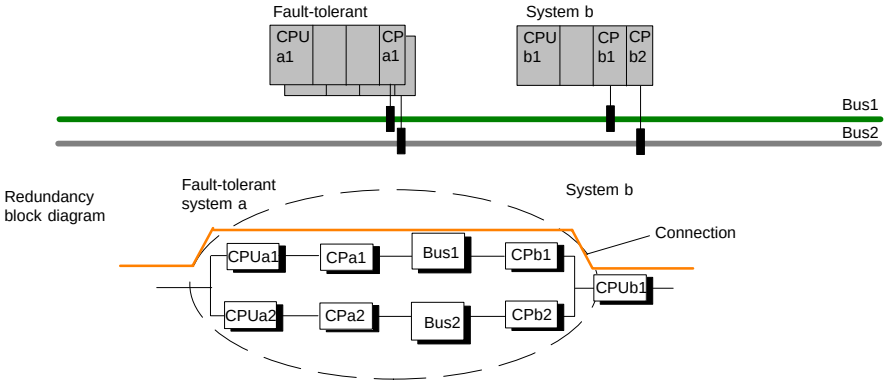


Figure 7-8 Example of Interconnected Standard and Fault-Tolerant Systems on a Redundant Bus System

Failure behavior

Two-fiber ring and bus systems

Since standard S7 connections are used in this particular instance (the connection terminates at the CPU of the subsystem, in this instance CPUa1), an error on the fault-tolerant system – for example, CPUa1 or CPa1 – and an error on system b – for example, CP b – both result in total failure of communications between the systems involved (refer to Figures 7-7 and 7-8).

There are no differences specific to the bus system for failure behavior.

7.5.2 Communications over Redundant S7 Connections

Availability

Availability can be enhanced by using a redundant system bus and by using two separate CPs on a standard system.

Redundant communications can be operated even with standard connections. Two separate S7 connections have to be configured for this. Connection redundancy has to be implemented by means of programming for this purpose. Communication monitoring has to be implemented for both connections at the user program level to detect a communications failure and to switch to the second connection.

Figure 7-9 shows an example of such a configuration.

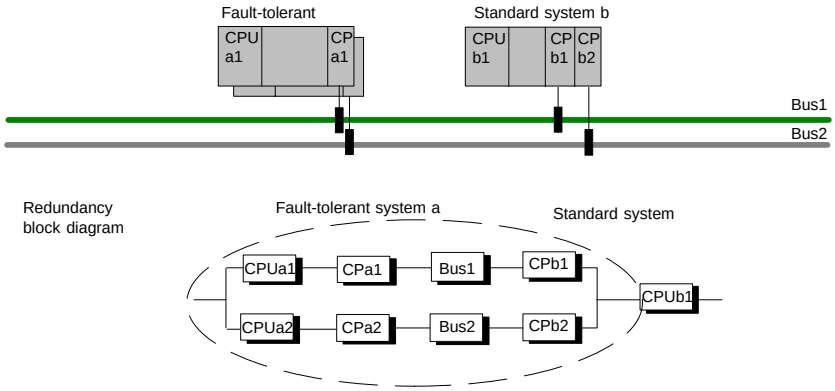


Figure 7-9 Example of Redundancy with Fault-Tolerant Systems and Redundant Bus System with Redundant Standard Connections

Failure behavior

Double errors on the fault-tolerant system – in other words, CPUa1 and CPa 2 – double errors on the standard system (CPb1 and CPb2) and a single error on the standard system (CPUb1) result in total failure of communications between the systems involved (refer to Figure 7-9).

7.5.3 Communications via a Point-to-Point CP on the ET200M

Connection via ET200M

Connections of fault-tolerant systems to single-channel systems are frequently possible only through a point-to-point connection since many systems have no other connection option.

To have the data of a single-channel system available on the CPUs of the fault-tolerant system as well, the point-to-point CP (CP 341) has to be inserted in a distributed mounting rack with two IM 153-2s.

Configuring connections

Redundant connections between the point-to-point CP and the fault-tolerant system are not necessary.

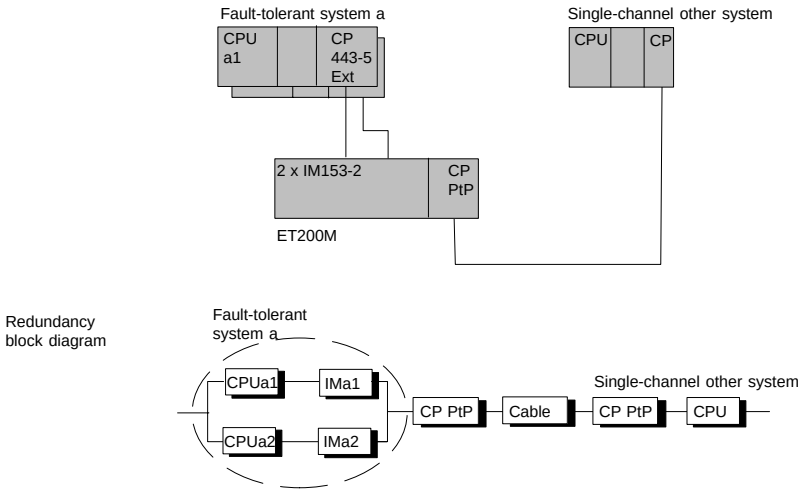


Figure 7-10 Example of Interconnection of a Fault-Tolerant System and a Single-Channel Non-Siemens System

Failure behavior

Double errors in the fault-tolerant system (i.e. CPUa1 and IM153-2) and single errors in the third-party system will result in total failure of communications between the systems involved (refer to Figure 7-10).

The point-to-point CP can alternatively be inserted centrally in “fault-tolerant system a”. But with this configuration even a failure of the CPU results in total failure of communications.

7.5.4 Random Connection with Single-channel Systems

Connection via a PC as gateway

When fault-tolerant systems are connected to single-channel systems, they can alternatively be connected via a gateway (no connection redundancy). The gateway is connected via one or two CPs to the system bus, depending on availability requirements. Fault-tolerant connections can be configured between the gateway and the fault-tolerant systems. The gateway makes it possible to link any kind of single-channel system – for example, TCP/IP with a specific manufacturer's protocol).

A software instance written by the user in the gateway implements the single-channel transition to the fault-tolerant systems. Any single-channel system can be connected in this way to a fault-tolerant system.

Configuring connections

Fault-tolerant connections are not required between the gateway CP and the single-channel system.

The gateway CP is located on a PC system that has fault-tolerant connections to the fault-tolerant system.

In order to be able to use fault-tolerant S7 connections between fault-tolerant system A and the gateway, the "S7-REDCONNECT" software package is required on the gateway. Data conversion for routing via the single-channel connection has to be implemented in the user program.

You will find further information on this point in the catalog called "*Industrial Communications IK10*".

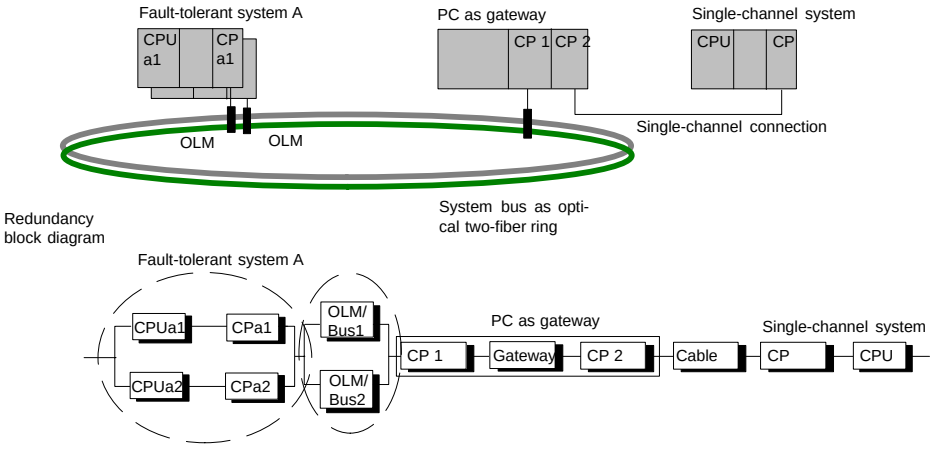


Figure 7-11 Example of Interconnection of a Fault-Tolerant System and a Single-Channel Non-Siemens System

Configuring with STEP 7

8

This chapter presents an overview of the special features and possibilities of the S7-400H options package.

The first section describes how to install the options package.

The second section lists the extensions of the STEP 7 options package and summarizes some central points which you have to take into account when you are configuring a fault-tolerant system.

The third section deals with the programming device functions contained in the STEP 7 options package.

You will find a more detailed description in basic Help dealing with options packages, *Configuring Fault-tolerant Systems*. You will find the Help under the menu option **Help > Help topics > Help on option packages**.

8.1 Installing the Options Package

Software requirements

In order to install the “S7 fault-tolerant system” option package, version 1 or higher, you must have the STEP 7 standard package, V5.1 (or higher) installed on your PG or PC.

Installing the options package

1. Start the PC or programming device on which you have installed the STEP 7 standard package and make sure that no STEP 7 applications are open.
2. Insert the product CD for the options package.
3. Call the SETUP.EXE program on the CD.
4. Follow the instructions given by the setup program and select the options you require.

Reading the Readme file

Important late-breaking information on the software supplied is recorded in a readme file. You can view this file when you have completed the Setup program or open it at a later point of time. It is located in the **S7hsys** directory of STEP 7.

Starting the options package

The options package does not contain any applications that have to be started explicitly. The additional options are integrated in the familiar user interface.

Displaying integrated Help

For the dialog boxes of the options package there is integrated Help, which you call at any stage of configuration either by pressing F1 or clicking the **Help** button. You can obtain more detailed information by choosing **Help > Help Topics** from the menu.

8.2 Configuring with STEP 7

The basic approach to configuring the S7-400H is no different from that used to configure the S7-400 – in other words

- creating projects and stations
- configuring hardware and networking
- loading system data onto the programmable logic controller.

Even the different steps that are required for this are identical for the most part to those with which you are familiar from the S7-400.

Note

On the S7-400H, you must always load the following error OBs onto the CPU: OB 70, OB 72, OB 80, OB 82, OB 83, OB 85, OB 86, OB 87, OB 121 and OB 122. If these OBs are not loaded, the fault-tolerant system goes to the Stop system mode in the event of an error.

Creating an H station

The SIMATIC H station is offered as a separate station type by SIMATIC Manager. It allows the configuration of two central racks, each having a CPU and thus the redundant H station configuration.

8.2.1 Rules for H Station Equipment

The following rules have to be complied with for an H station, in addition to the rules that generally apply to the arrangement of modules in the S7-400:

- The central processing units must be inserted in the same slots in each case.
- Redundantly used external DP master interfaces or communication modules must be inserted in the same slots in each case.
- External DP master interface modules for redundant DP master systems should only be inserted in central racks and not in expansion racks.
- Redundantly used modules (for example, CPU 417-4H, DP slave interface module IM 153-2) must be identical – in other words, they must have the same order number and the same version, and the same firmware version.

Installation rules

- An H station may contain up to 20 expansion racks.
- Even-numbered mounting racks can be assigned only to central controller 0, whereas odd-numbered mounting racks can be assigned only to central controller 1.
- Modules connected to a communication bus can be operated only in mounting racks 0 through 6.
- Communication-bus capable modules are not permissible in switched I/Os.
- Pay attention to the mounting rack numbers when operating CPs for fault-tolerant communications in expansion racks:

  The numbers must be directly sequential and begin with the even number – for example, mounting racks numbers 2 and 3, but not mounting racks numbers 3 and 4.
- A mounting rack number is also assigned for DP master No. 9 onwards when equipping a central controller with DP master modules. The number of possible expansion racks is reduced as a result.

Compliance with the rules is monitored automatically by STEP 7 and taken into account in an appropriate manner during configuration.

8.2.2 Configuring Hardware

The simplest way of achieving a redundant hardware configuration consists in initially equipping **one** mounting rack fully with all the components required to be redundant, assigning parameters to them and then copying them.

You can then specify the various addresses (for one-way I/O only!) and arrange other, non-redundant modules in individual racks.

Special features in presenting the hardware configuration

To make it possible for a redundant DP master system to be detected quickly, it is represented by two closely parallel DP cables.

8.2.3 Configuring Networks

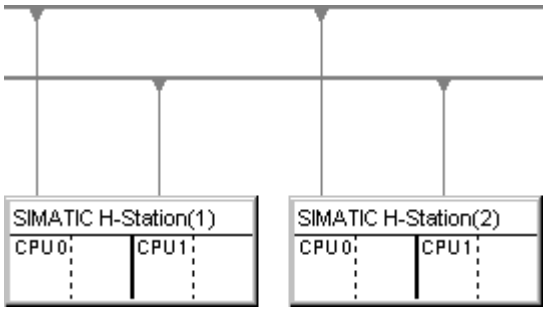
The fault-tolerant S7 connection is a separate connection type of the “Configure Networks” application. The following communication peers can communicate with each other:

- S7 H station (with 2 H-CPUs) –> S7 H station (with 2 H-CPUs)
- S7 400 station (with 1 H-CPU) –> S7 H station (with 2 H-CPUs)
- SIMATIC PC stations –> S7 H station (with 2 H-CPUs)

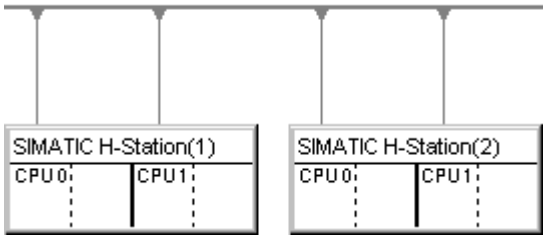
Fault-tolerant connections are only permissible if at least one communications terminal is an S7 H station.

When this type of connection is being configured, the application automatically determines the number of possible connection paths:

- If two independent but identical subnets are available which are both suitable for an S7 connection (DP master systems), two connection paths will be used. In practice these are generally electrical networks, each a CP in a subnet:



- If only one DP master system is available – in practice typically fiber-optic cables – four connection paths are used for a connection between two H stations. All the CPs are in this subnet:



Downloading the network configuration to the H station

The network configuration can be downloaded to the entire H station in one go. To do this, the same requirements must be met as for downloading the network configuration to a standard station.

8.3 Programming Device Functions in STEP 7

Display in SIMATIC Manager

In order to do justice to the special features of an H station, the way in which the system is displayed and edited in SIMATIC Manager differs from that of a S7-400 standard station as follows:

- In the offline view, the S7 program is displayed only under CPU0 of the H station. No S7 program is visible under CPU1.
- In the online view, the S7 program is displayed under both central processing units and can be selected in both locations.

Communication functions

For programming device functions that lead to the establishment of an online connection – for example, downloading and deleting blocks – one of the two CPUs has to be selected even if the function affects the entire system over the redundant link.

- Data which are modified in one of the central processing units in redundant operation affect the other CPUs over the redundant link.
- Data which are modified when there is no redundant link – in other words, in Solo mode – initially affect only the edited CPU. The blocks are applied by the master CPU to the standby CPU during the next link-up and update procedure. Exception: after a configuration modification no new blocks are applied (only the unchanged data blocks). Loading the blocks is then the responsibility of the user.

Failure and Replacement of Components During Operation

9

One factor that is crucial to the uninterrupted operation of the fault-tolerant controller is the replacement of failed components while in operation. Rapid repair quickly reestablishes the fault tolerance.

We will show you in the sections that follow how simple and fast it can be to repair and replace components in the S7-400H. Please pay attention to the tips in the corresponding sections of the installation manual, *S7-400/M7-400 Programmable Controllers, Hardware and Installation*

9.1 Failure and Replacement of Components in Central Racks and Expansion Racks

Which components can be replaced?

The following components can be replaced during operation:

- central processing units – for example, CPU 417-4H
- power supply modules – for example, PS 405 and PS 407
- signal and function modules
- communication processors
- synchronization submodules and fiber-optic cables
- interface modules – for example, IM 460 and IM 461

9.1.1 Failure and Replacement of a Central Processing Unit CPU 417-4H

Complete replacement of the CPU is not always necessary. If the failure affects only the load memory, all you have to do is replace the memory card concerned. Both cases are described below.

Starting situation for replacement of the complete CPU

Failure	How Does the System React?
The S7-400H is in the Redundant system mode and a **CPU** fails.	• Partner CPU switches to Solo mode. • Partner CPU reports the event in the diagnostic buffer and via OB 72.

Requirements for a replacement

The module replacement described below is possible only if the “new” central processing unit

- has the same operating system status as the failed CPU (refer to Operating system update in section 10.8) und
- has the same main memory and load memory as the failed CPU.

Procedure

To change a central processing unit, perform the following steps:

Step	What Has To Be Done?	How Does the System React?
1	Turn off the power supply module.	• Entire subsystem is turned off (system operating in Solo mode).
2	Replace the central processing unit.	-
3	Plug in the synchronization submodules. Make sure the rack number is set correctly.	-
4	Plug in the fiber-optic cable connections of the synchronization submodules.	-
5	Turn the power supply back on.	• CPU executes the self-tests and goes to STOP.
6	Perform Memory Reset on the replaced CPU.	-
7	Start the replaced CPU – for example, STOP→RUN or Start via programming device.	• CPU performs automatic LINK-UP and UPDATE. • CPU changes to RUN and operates as the standby CPU.

Starting situation for replacement of the load memory

Failure	How Does the System React?
The S7-400H is in the Redundant system mode and an **error access to the load memory** is executed.	• Affected CPU switches to STOP and makes a reset request. • Partner CPU switches to Solo mode.

Procedure

To change the load memory perform the following steps:

Step	What Has To Be Done?	How Does the System React?
1	Change the memory card on the stopped CPU.	-
2	Perform a reset on the CPU with the replaced memory card.	-
3	Start the CPU.	• CPU performs automatic LINK-UP and UPDATE. • CPU changes to RUN and operates as the standby CPU.

9.1.2 Failure and Replacement of a Power Supply Module

Initial situation

Both central processing units are at RUN.

Failure	How Does the System React?
The S7-400H is in the Redundant system mode and one power supply module fails.	• Partner CPU switches to Solo mode. • Partner CPU reports the event in the diagnostic buffer and via OB 72.

Procedure

To change a power supply module in the central rack, perform the following steps:

Step	What Has To Be Done?	How Does the System React?
1	Turn off the power supply (24 V DC for PS 405 or 120/230 V AC for PS 407).	• Entire subsystem is turned off (system operating in Solo mode).
2	Replace the module.	-
3	Turn the power supply back on.	• CPU executes the self-tests. • CPU performs automatic LINK-UP and UPDATE. • CPU changes to RUN (Redundant system mode) and now operates as the standby CPU.

Note

If using a redundant power supply PS 407 10A R two power supply modules are assigned to a CPU 417-4H. If a part of the redundant PS 407 10A R power supply module fails, the corresponding CPU keeps on running. The defective part can be replaced during operation.

Other power supply modules

If the failure concerns a power supply module outside the central rack (e.g. in the expansion rack or in the I/O device) the failure is reported as a rack failure (central) or station failure (remote). In this case, simply switch off the power supply to the power supply module concerned.

9.1.3 Failure and Replacement of an Input/Output or Function Module

Initial situation

Failure	How Does the System React?
The S7-400H is in the Redundant system mode and an input/output or function module fails.	• Both CPUs report the event in the diagnostic buffer and via appropriate OBs.

Procedure

To replace signal and function modules (central or remote), perform the following steps:

Step	What Has To Be Done?	How Does the System React?
1	Disconnect the wiring.	• Call OB 82 if the module concerned is diagnosis-interruptible and diagnostic interrupts are released via the configuration. • – Call OB 122 if you are accessing the module by direct access – Call OB 85 if you are accessing the module by means of the process image
2	Extract the failed module (in RUN mode).	• Both CPUs process the insert/remove-module interrupt OB 83 in synchronism.
3	Insert the new module.	• Both CPUs process the insert/remove-module interrupt OB 83 in synchronism. • Parameters are assigned automatically to the module by the CPU concerned and the module is addressed again.
4	Connect the wiring.	Call OB 82 if the module concerned is diagnosis-interruptible and diagnostic interrupts are released via the configuration.

9.1.4 Failure and Replacement of a Communication Processor

This section describes the failure and replacement of communication processors for the PROFIBUS and Industrial Ethernets.

The failure and replacement of communication processors for the PROFIBUS-DP are described in Section 9.2.1.

Initial situation

Failure	How Does the System React?
The S7-400H is in the Redundant system mode and one communication processor fails.	• Both CPUs report the event in the diagnostic buffer and via appropriate OBs. • With communications via standard connections Connection failed • With communications via redundant connections Communications are maintained without interruption over an alternative channel.

Procedure

To replace a communication processor for a PROFIBUS or an industrial Ethernet, perform the following steps:

Step	What Has To Be Done?	How Does the System React?
1	Extract the module.	Both CPUs process the insert/remove-module interrupt OB 83 in synchronism.
2	Make sure that the new module has no parameterization data in its integrated FLASH EPROM and plug it in.	• Both CPUs process the insert/remove-module interrupt OB 83 in synchronism. • The module is automatically parameterized by the appropriate CPU.
3	Turn the module back on.	The module resumes communications (system establishes communication connection automatically).

9.1.5 Failure and Replacement of a Synchronization Submodule or Fiber-Optic Cable

In this section three different error scenarios are to be differentiated:

- Failure of a synchronization submodule or fiber-optic cable
- Successive failure of the two synchronization submodules or fiber-optic cables
- Simultaneous failure of the two synchronization submodules or fiber-optic cables

Initial situation

Failure	How Does the System React?
Failure of a synchronization submodule or fiber-optic cable: The S7-400H is in the Redundant system mode and a fiber-optic cable or a synchronization submodule fails.	• Master CPU reports the event in the diagnostic buffer and via OB 72. • Master CPU remains at RUN; standby CPU goes to STOP

Procedure

To replace a synchronization submodule or fiber-optic cable, perform the following steps:

Step	What Has To Be Done?	How Does the System React?
1	Replace the fiber-optic cable first. [1]	-
2	Start the standby CPU – for example, STOP→RUN or Start via programming device.	The following reactions are possible: 1. CPU goes to RUN mode. 2. CPU goes to STOP mode. In this case continue with step 3.
3	Unplug the faulty synchronization submodule from the standby CPU. To do so, screw the threaded pin in the additional front panel of the synchronization submodule into the thread of the module.	-
4	Insert the new synchronization submodule in the standby CPU.[1] Make sure the rack number is set correctly.	-
5	Plug in the fiber-optic cable connections of the synchronization submodules.	-
6	Start the standby CPU – for example, STOP→RUN or Start via programming device.	The following reactions are possible: 1. CPU goes to RUN mode. 2. CPU goes to STOP mode. In this case continue with step 7.

Step	What Has To Be Done?	How Does the System React?
7	If in step 6 the standby CPU has gone to STOP: Extract the synchronization submodule from the master CPU.	• Master CPU executes insert/remove-module interrupt OB 83 and redundancy error OB 72 (incoming).
8	Plug the new synchronization submodule into the master CPU. Make sure the rack number is set correctly.	• Master CPU executes insert/remove-module interrupt OB 83 and redundancy error OB 72 (outgoing).
9	Plug in the fiber-optic cable connections of the synchronization submodules.	-
10	Start the standby CPU – for example, STOP→RUN or Start via programming device.	• CPU performs automatic LINK-UP and UPDATE. • CPU changes to RUN (Redundant system mode) and now operates as the standby CPU.

1 The CPU displays by means of LEDs and by means of the diagnosis whether the lower or upper redundant link has failed. After the defective parts (fiber-optic cable or synchronization submodule) have been replaced, LEDs IFM1F and IFM2F go out. Not until then can you perform the next step.

Note

If both fiber-optic cables or synchronization submodules are successively damaged or replaced the system reactions are the same as described above.

The only exception is that the standby CPU does not go to STOP mode but instead requests a reset.

Initial situation

Failure	How Does the System React?
Simultaneous failure of the two synchronization submodules or fiber-optic cables: The S7-400H is in the Redundant system mode and **both** fiber-optic cables or synchronization submodules fail.	• Both CPUs report the event in the diagnostic buffer and via OB 72. • Both CPUs become the master CPU and remain in RUN mode.

Procedure

The double error described results in loss of redundancy. In this event proceed as follows:

Step	What Has To Be Done?	How Does the System React?
1	Switch off a subsystem.	-
2	Replace the faulty components.	-
3	Turn the subsystem back on.	LEDs IFM1F and IFMF2F go out. The standby LED lights.
4	Start the CPU – for example, STOP→RUN or Start via programming device.	• CPU performs automatic LINK-UP and UPDATE. • CPU changes to RUN (Redundant system mode) and now operates as the standby CPU.

9.1.6 Failure and Replacement of an IM 460 and IM 461 Interface Module

The IM 460 and IM 461 interface modules make it possible to connect expansion racks.

Initial situation

Failure	How Does the System React?
The S7-400H is in the Redundant system mode and one interface module fails.	• Connected expansion unit is turned off. • Both CPUs report the event in the diagnostic buffer and via OB 86.

Procedure

To change an interface module, perform the following steps:

Step	What Has To Be Done?	How Does the System React?
1	Turn off the power supply of the central rack.	• The partner CPU switches to Solo mode.
2	Turn off the power supply of the expansion unit in which you want to replace the interface module.	-
3	Extract the interface module.	-
4	Insert the new interface module and turn the power supply of the expansion unit back on.	-
5	Switch the power supply of the central rack back on and start the CPU.	• CPU performs automatic LINK-UP and UPDATE. • CPU changes to RUN and operates as the standby CPU.

9.2 Failure and Replacement of Components of the Distributed I/O

Which components can be replaced?

The following components of the distributed I/O can be replaced during operation:

- PROFIBUS-DP master
- PROFIBUS-DP interface module (IM 153-2 or IM 157)
- PROFIBUS DP slave
- PROFIBUS-DP cable

Note

Changing modules in a remote station has been described previously in Section 9.1.3.

9.2.1 Failure and Replacement of a PROFIBUS-DP Master

Initial situation

Failure	How Does the System React?
The S7-400H is in the Redundant system mode and one DP master module fails.	• With single-channel, one-sided I/O: DP master can no longer process connected DP slaves. • With switched I/O: DP slaves are addressed via the DP master of the partner.

Procedure

To replace a PROFIBUS-DP master, perform the following steps:

Step	What Has To Be Done?	How Does the System React?
1	Turn off the power supply of the central rack.	The fault-tolerant system goes to Solo mode.
2	Unplug the DP Profibus cable for the affected DP master module.	-
3	Replace the affected module.	-
4	Plug in the DP Profibus cable again.	-
5	Turn on the power supply of the central rack.	• CPU performs automatic LINK-UP and UPDATE. • CPU changes to RUN and operates as the standby CPU.

9.2.2 Failure and Replacement of a redundant PROFIBUS-DP Interface Module

Initial situation

Failure	How Does the System React?
The S7-400H is in the Redundant system mode and a PROFIBUS-DP interface module (IM 153-2, IM 157) fails.	Both CPUs report the event in the diagnostic buffer and via OB 70.

Replacement procedure

To replace PROFIBUS-DP interface module, perform the following steps:

Step	What Has To Be Done?	How Does the System React?
1	Turn off the supply for the affected DP interface module.	-
2	Unplug the connected bus connector.	-
3	Insert the new DP Profibus interface module and turn the supply back on.	-
4	Plug the bus connector back on.	• CPUs process the mounting rack failure OB 70 in synchronism (event having the state cleared). • Redundant access to the station is again possible for the system.

9.2.3 Failure and Replacement of a PROFIBUS-DP Slave

Initial situation

Failure	How Does the System React?
The S7-400H is in the Redundant system mode and one DP slave fails.	Both CPUs report the event in the diagnostic buffer and via the appropriate OB.

Procedure

To replace a DP slave, perform the following steps:

Step	What Has To Be Done?	How Does the System React?
1	Turn off the supply for the DP slave.	-
2	Unplug the connected bus connector.	-
3	Replace the DP slave.	-
4	Plug the bus connector on and turn the supply back on.	• CPUs process the mounting rack failure OB 86 in synchronism (event having the state cleared). • DP slave can be addressed by the relevant DP master system.

9.2.4 Failure and Replacement of PROFIBUS-DP Cables

Initial situation

Failure	How Does the System React?
The S7-400H is in the Redundant system mode and the PROFIBUS-DP cable is defective.	• With single-channel, one-sided I/O: Rack failure OB (OB 86) is started (incoming event). DP master can no longer process connected DP slaves (station failure). • With switched I/O: I/O redundancy error OB (OB 70) is started (incoming event). DP slaves are addressed via the DP master of the partner.

Replacement procedure

To replace PROFIBUS-DP cables, perform the following steps:

Step	What Has To Be Done?	How Does the System React?
1	Check the wiring and localize the interrupted PROFIBUS-DP cable.	-
2	Replace the defective cable.	-
3	Switch the failed modules to RUN.	CPUs process error OBs in synchronism • With one-sided I/O: Mounting rack failure OB 86 (event having the state cleared) DP slaves can be addressed via the DP master system. • With switched I/O: I/O redundancy error OB70 (event having the state cleared). DP slaves can be addressed via both DP master systems.

Modifications to the System while in Operation

10

In addition to the possibilities described in Chapter 9 for replacing failed components while in operation, CPU 417-4H firmware version V2.0.0 or higher also allows a system modification to be performed without interrupting the program running.

This procedure is partly dependent on whether you are executing your user program in PCS7 or in STEP 7.

The procedures described below for making modifications while in operation are each structured so as to start from the Redundant system mode (see section 4.2) and aim to return to this again.

Note

Keep strictly to the rules described in this chapter with regard to modifications of the system during routine operation. If you contravene one or more rules, the response of the fault-tolerant system can result in its availability being restricted or even failure of the entire programmable logic controller.

Safety-relevant components are not taken into account in this description. For more details on dealing with the fail-safe systems refer to the manual *S7-400F/S7-400FH Programmable Controllers, Fail-Safe Systems*.

10.1 Possible Hardware Modifications

How is a hardware modification performed?

If the hardware components concerned are suitable for unplugging or plugging in live the hardware modification can be carried out in the Redundant system mode. However, since loading a modified hardware configuration in the Redundant system mode would result in the fault-tolerant system stopping this must temporarily be put into Solo mode. In Solo mode the process is then controlled by only one CPU while the desired configuration modifications are carried out on the other CPU.

Note

You should load configuration changes into the CPU only from “Configure Hardware”.

Since in this process the load memory content of both CPUs has to be modified more than once, expansion of the integrated load memory with a RAM card is to be recommended (at least temporarily) is recommended.

You may only perform the substitution of the FLASH card with required for this with a RAM card if the FLASH card has as much storage space at most as the largest RAM card available. If your FLASH card is larger than the largest available RAM card, you must perform the necessary configuration and program changes in such small steps that there is space for them in the integrated load memory.

Caution

Whenever making hardware modifications make sure that the synchronization link between the two CPUs is re-established **before** the standby CPU is started or activated. When the power supplies of the CPUs are switched on LEDs IFM1F and IFM2F – which are used to indicate faults on the interfaces for memory submodules – must **go out** on both CPUs.

Which components can be modified?

The following modifications can be made to the hardware configuration during operation:

- Adding or removing modules to/from the central or expansion units (e.g. one-way I/O module).

Note

The addition or removal of interface modules IM460 and IM461, external DP master interface module CP443-5 Extended and the associated connecting cables is not allowed.

- Adding or removing components of the remote input/output station, such as
  - DP slaves with redundant interface module (e.g. ET 200M or PA link)
  - one-way DP slaves (in any DP master system)
  - modules in modular DP slaves
  - PA coupler
  - PA devices
- Use of a free channel on an existing module
- Changing certain CPU parameters
- Changing the Memory Components of the CPU

With all modifications observe the rules for the configuration of an H station (see Section 8.2.1).

What should I note at the system planning stage?

In order for switched I/O to be able to be expanded while in operation the following points are to be taken into account at the system planning stage:

- In both cables of redundant DP master system sufficient numbers of branching points are to be provided for spur lines or dividing points (spur lines are not permissible at transmission rates of 12 MBd). These may either be provided at regular intervals or at all well accessible points.
- Both cables are to be uniquely identified so that the line which is currently active is not accidentally cut off. This identification should be visible not only at the end points of a line but also at each possible new connection point. Different colored cables are excellent for this.
- Modular DP slave stations (ET 200M) and DP-PA links are always to be built up with an active backplane bus and fitted as fully as possible with bus units since the bus units cannot be plugged in and unplugged while in operation.
- PROFIBUS DP and PROFIBUS PA LAN cables are to be equipped with active bus terminators at both ends so that the lines continue to be correctly terminated during the modification work.
- PROFIBUS PA bus systems should be built up using components from the SpliTConnect product range (see interactive catalog CA01) so that separation of the lines is not required.

- Loaded data blocks must not be cleared and generated again, i.e. the SFCs 22 (CREATE_DB) and 23 (DEL_DB) may not be applied to DB numbers occupied by loaded DBs.
- Make sure that at the time the system modification is made on the PG/ES the current status of the user program is still available as a STEP 7 project in modular form. It is not sufficient for the user program from one of the CPUs to be loaded back into the PG/ES or compiled again from a STL source file.

Modification of the hardware configuration

With a few exceptions, all segments of the configuration can be modified while in operation. As a rule, a configuration modification also results in a modification in the user program.

The following must not be modified:

- certain CPU parameters (for details refer to the relevant subsections)
- the transmission rate (baud rate) of redundant DP master systems
- S7 and S7H connections

Modifications to the user program and the connection configuration

The modifications to the user program and the connection configuration are loaded into the PLC in the Redundant system mode. The procedure depends on the software used. For more details refer to the manuals under *Programming with STEP 7 V5.1* und *PCS 7, Configuration Manual*.

Special Features

- Do not carry out more modifications than you can keep an overview of. We recommend that you modify only one DP master and/or a few DP slaves (e.g. no more than 5) per reconfiguration run.
- During routine operation, you can add or remove modules in DP stations with a redundant PROFIBUS-DP interface module only with the interface modules IM 153-2, IM 153-2FO or IM 157 specified in Section 6.3.
- In the case of IM 153-2 bus modules can only be plugged in is the power supply is interrupted.

Preparations

To keep the time during which the fault-tolerant system must run in Solo mode to a minimum you should perform the following steps **before** commencing the hardware modification:

- Make sure that the memory components of the CPUs are sufficient for the new configuration and the new user program. If necessary, first expand the memory components (see section 10.7).
- Make sure that modules that are plugged in but not configured do not have any effect on the process.

10.2 Adding Components in PCS7

Initial situation

You have ensured that the CPU parameters (e.g. the monitoring times) suit the planned new program. If necessary you must first change the CPU parameters accordingly (see Section 10.6).

The fault-tolerant system is working in the Redundant system mode.

Procedure

In order to add hardware components to a fault-tolerant system under PCS7 the steps listed below are to be performed. Details of each step are listed in a subsection.

Step	What Has To Be Done?	Refer to Section
1	Modification of Hardware	10.2.1
2	Offline Modification of the Hardware Configuration	10.2.2
3	Stopping the Standby CPU	10.2.3
4	Loading new Hardware Configuration in the Standby CPU	10.2.4
5	Switch to CPU with modified configuration	10.2.5
6	Transition to the Redundant System Mode	10.2.6
7	Changing and Loading User Program	10.2.7

Exception

This overall process of system modification does not apply to the use of free channels on an existing module (see Section 10.2.8).

10.2.1 PCS7, Step 1: Modification of Hardware

Initial situation

The fault-tolerant system is working in the Redundant system mode.

Procedure

1. Add the new components to the system.
   - Plug new central modules into the rack.
   - Plug new module into existing modular DP stations.
   - Add new DP stations to existing DP master systems.

Note

With switched I/O: End all modifications on **one** line of the redundant DP master system first before starting modifications to the second line.

2. Connect the required sensors and actuators to the new components.

Result

Plugging in modules that have not yet been configured will have no effect on the user program. The same applies to adding DP stations.

The fault-tolerant system continues to work in the Redundant system mode.

New components are not yet addressed.

10.2.2 PCS7, Step 2: Offline Modification of the Hardware Configuration

Initial situation

The fault-tolerant system is working in the Redundant system mode.

Procedure

1. Perform all the modifications to the hardware configuration relating to the added hardware offline. Assign appropriate symbols to the new channels to be used.
2. Compile the new hardware configuration, but do **not** load it into the PLC yet.

Result

The modified hardware configuration is in the PG/ES. The PLC continues to work with the old configuration in the Redundant system mode.

Configuring connections

Connections from or to newly added CPs must be configured for both connection partners **after** modification of the hardware configuration is complete.

10.2.3 PCS7, Step 3: Stopping the Standby CPU

Initial situation

The fault-tolerant system is working in the Redundant system mode.

Procedure

1. In SIMATIC Manager select a CPU of the fault-tolerant system and select the menu command **PLC > Operating Mode**.
2. In the **Operating Mode** dialog box select the standby CPU and click the **Stop** button.

Result

The standby CPU switches to STOP mode, the master CPU remains in RUN mode, the fault-tolerant system works in Solo mode. One-way I/O of the standby CPU are no longer addressed.

Whilst I/O access errors of the one-way I/O will result in OB 85 being called, due to the superior CPU redundancy loss (OB 72) these will not be reported. OB 70 (I/O redundancy loss) will not be called.

10.2.4 PCS7, Step 4: Loading new Hardware Configuration in the Standby CPU

Initial situation

The fault-tolerant system is working in Solo mode.

Procedure

Load the compiled hardware configuration in the standby CPU that is in STOP mode.

Note

The user program and the connection configuration must not be overloaded in Solo mode.

Result

The new hardware configuration of the standby CPU does not yet have an effect on current operation.

10.2.5 PCS7, Step 5: Switch to CPU with modified configuration

Initial situation

The modified hardware configuration is loaded into the standby CPU.

Procedure

1. In SIMATIC Manager select a CPU of the fault-tolerant system and select the menu command **PLC > Operating Mode**.
2. In the **Operating Mode** dialog box click the **Switch to button.**
3. In the **Switch to** dialog box select the option **with modified configuration** and click the **Switch** button.
4. Confirm the check question that follows with **OK**.

Result

The standby CPU connects, is updated (see Chapter 5) and becomes the master. The former master CPU switches to STOP mode, the fault-tolerant system works with the new hardware configuration in Solo mode.

Behavior of the I/O

Type of I/O	One-way I/O of previous master CPU	One-way I/O of new master CPU	Switched I/O
Added I/O modules	Are not addressed by the CPU.	Are parameterized and updated by the CPU. Driver blocks are not yet present. Any process or diagnostic interrupts occurring will be recognized but not reported.	
I/O modules that continue to be present	Are no longer addressed by the CPU. Output modules output the configured substitute or holding values.	Are re-parameterized [1] and updated by the CPU.	Continue to work without interruption.
Added DP stations	Are not addressed by the CPU.	as for added I/O modules (see above)	

1) Central modules are reset first in addition. Output modules briefly output 0 during this time (instead of the configured substitute or holding values).

Reaction if monitoring times are exceeded

If one of the monitored times exceeds the maximum value configured the update is interrupted and no change of master takes place. The fault-tolerant system remains in Solo mode with the previous master CPU and in certain conditions attempts to perform the change of master later. For more details refer to section 5.3.

10.2.6 PCS7, Step 6: Transition to the Redundant System Mode

Initial situation

The fault-tolerant system works with the new hardware configuration in Solo mode.

Procedure

1. In SIMATIC Manager select a CPU of the fault-tolerant system and select the menu command **PLC > Operating Mode**.
2. In the **Operating Mode** dialog box select the standby CPU and click the **Restart (warm restart)** button.

Result

Standby CPU links up again and is updated. The fault-tolerant system works with the new hardware configuration in the Redundant system mode.

Behavior of the I/O

Type of I/O	One-way I/O of standby CPU	One-way I/O of master CPU	Switched I/O
Added I/O modules	Are parameterized and updated by the CPU. Driver blocks are not yet present. Any interrupts occurring are not reported.	Are updated by the CPU. Driver blocks are not yet present. Any process or diagnostic interrupts occurring will be recognized but not reported.	
I/O modules that continue to be present	Are re-parameterized [1] and updated by the CPU.	Continue to work without interruption.	
Added DP stations	as for added I/O modules (see above)	Driver blocks are not yet present. Any interrupts occurring are not reported.	

1) Central modules are reset first in addition. Output modules briefly output 0 during this time (instead of the configured substitute or holding values).

Behavior if monitoring times are exceeded

If one of the monitored times exceeds the maximum value configured the update is interrupted. The fault-tolerant system remains in Solo mode with the previous master CPU and in certain conditions attempts to perform the link-up and update later. For more details refer to section 5.3.

10.2.7 PCS7, Step 7: Changing and Loading User Program

Initial situation

The fault-tolerant system works with the new hardware configuration in the Redundant system mode.

Caution

The following program modifications are not possible in the Redundant system mode and result in the system mode Stop (both CPUs in STOP mode):

- structural modifications to an FB interface or the FB instance data
- structural modifications to global DBs
- compression of the CFC user program.

Before the entire program is recompiled and reloaded due to such modifications the parameter values must be read back into the CFC, since otherwise the modifications to the block parameters could be lost. More details of this can be found in the manual *CFC for S7, Continuous Function Chart*.

Procedure

1. Perform all the program modifications relating to the added hardware. You can add the following components:
   - CFC and SFC charts
   - blocks in existing charts
   - connections and parameterizations
2. Assign parameters to the added channel drivers and connect these to the newly assigned symbols (see Section 10.2.2).
3. In SIMATIC Manager select the charts folder and select the menu command **Extras > Charts > Generate module drivers.**
4. Compile only the modifications in the charts and load these into the PLC.

Note

Until the first FC is called the value of its output is undefined. This is to be taken into account in the connection of the FC outputs.

5. Configure the connections from or to the newly added CPs on both connection partners and load these into the PLC.

Result

The fault-tolerant system processes the entire system hardware with the new user program in the Redundant system mode.

10.2.8 Use of free channels on an existing module

The use of previously free channels of an I/O module depends primarily on whether or not parameters can be assigned to the module.

Modules to which parameters cannot be assigned

In the case of modules to which parameters cannot be assigned free channels can be connected and used in the user program at any time.

Modules to which parameters can be assigned

In the case of modules to which parameters can be assigned the hardware configuration must be modified to the sensors or actuators to be used. This generally necessitates reassignment of parameters to entire module.

It is then no longer possible to operate the module concerned without interruption.

- One-way output modules briefly output 0 (instead of the configured substitute or holding values).
- New parameters are not assigned to modules in switched DP stations on switching to the CPU with the modified configuration. To change the channel usage the following procedure is required:
  - In steps I to VI (see Section 10.3) the module concerned is completely removed from the hardware configuration and the user program. However, it can remain plugged into the DP station. The module drivers must not be removed.
  - In steps 2 to 7 (see Section 10.2) the module is added to the hardware configuration and the user program again with the changed use.

Warning

Modules concerned will not be addressed between the two switching processes (steps V and 5); affected output modules output the value 0. The existing channel drivers in the user program retain their signals.

If this reaction cannot be tolerated for the process to be controlled then it is not possible to use previously free channels. In this case you must plug in additional modules in order to expand the system.

10.3 Removing Components in PCS7

Initial situation

You have ensured that the CPU parameters (e.g. the monitoring times) suit the planned new program. If necessary you must first change the CPU parameters accordingly (see Section 10.6).

The modules to be removed and the associated sensors and actuators are no longer of any significance for the process to be controlled. The fault-tolerant system is working in the Redundant system mode.

Procedure

In order to remove hardware components from a fault-tolerant system under PCS7 the steps listed below are to be performed. Details of each step are listed in a subsection.

Step	What Has To Be Done?	Refer to Section
I	Offline Modification of the Hardware Configuration	10.3.1
II	Changing and Loading User Program	10.3.2
III	Stopping the Standby CPU	10.3.3
IV	Loading new Hardware Configuration in the Standby CPU	10.3.4
V	Switch to CPU with modified configuration	10.3.5
VI	Transition to the Redundant System Mode	10.3.6
VII	Modification of Hardware	10.3.7

10.3.1 PCS7, Step I: Offline Modification of the Hardware Configuration

Initial situation

The fault-tolerant system is working in the Redundant system mode.

Procedure

1. Perform offline only the configuration modifications relating to the hardware to be removed. As you do, delete the symbols to the channels that are no longer used.
2. Compile the new hardware configuration, but do **not** load it into the PLC yet.

Result

The modified hardware configuration is in the PG/ES. The PLC continues to work with the old configuration in the Redundant system mode.

10.3.2 PCS7, Step II: Changing and Loading User Program

Initial situation

The fault-tolerant system is working in the Redundant system mode.

Caution

The following program modifications are not possible in the Redundant system mode and result in the system mode Stop (both CPUs in STOP mode):

- structural modifications to an FB interface or the FB instance data
- structural modifications to global DBs
- compression of the CFC user program.

Before the entire program is recompiled and reloaded due to such modifications the parameter values must be read back into the CFC, since otherwise the modifications to the block parameters could be lost. More details of this can be found in the manual *CFC for S7, Continuous Function Chart*.

Procedure

1. Perform only the program modifications relating to the hardware to be removed. You can remove the following components:
   - CFC and SFC charts
   - blocks in existing charts
   - channel drivers, connections and parameterizations
2. In SIMATIC Manager select the charts folder and select the menu command **Extras > Charts > Generate module drivers.**

   This removes the driver blocks that are no longer required.
3. Compile only the modifications in the charts and load these into the PLC.

Note

Until the first FC is called the value of its output is undefined. This is to be taken into account in the connection of the FC outputs.

Result

The fault-tolerant system continues to work in the Redundant system mode. The modified user program will no longer attempt to access the hardware to be removed.

10.3.3 PCS7, Step III: Stopping the Standby CPU

Initial situation

The fault-tolerant system is working in the Redundant system mode. The user program will no longer attempt to access the hardware to be removed.

Procedure

1. In SIMATIC Manager select a CPU of the fault-tolerant system and select the menu command **PLC > Operating Mode**.
2. In the **Operating Mode** dialog box select the standby CPU and click the **Stop** button.

Result

The standby CPU switches to STOP mode, the master CPU remains in RUN mode, the fault-tolerant system works in Solo mode. One-way I/O of the standby CPU are no longer addressed.

10.3.4 PCS7, Step IV: Loading new Hardware Configuration in the Standby CPU

Initial situation

The fault-tolerant system is working in Solo mode.

Procedure

Load the compiled hardware configuration in the standby CPU that is in STOP mode.

Note

The user program and the connection configuration must not be overloaded in Solo mode.

Result

The new hardware configuration of the standby CPU does not yet have an effect on current operation.

10.3.5 PCS7, Step V: Switch to CPU with modified configuration

Initial situation

The modified hardware configuration is loaded into the standby CPU.

Procedure

1. In SIMATIC Manager select a CPU of the fault-tolerant system and select the menu command **PLC > Operating Mode**.
2. In the **Operating Mode** dialog box click the **Switch to button.**
3. In the **Switch to** dialog box select the option **with modified configuration** and click the **Switch** button.
4. Confirm the check question that follows with **OK**.

Result

The standby CPU connects, is updated (see Chapter 5) and becomes the master. The former master CPU switches to STOP mode, the fault-tolerant system works with the new hardware configuration in Solo mode.

Behavior of the I/O

Type of I/O	One-way I/O of previous master CPU	One-way I/O of new master CPU	Switched I/O
I/O modules to be removed[1)]	Are no longer addressed by the CPU. Driver blocks are no longer present.		
I/O modules that continue to be present	Are no longer addressed by the CPU. Output modules output the configured substitute or holding values.	Are re-parameterized [2)] and updated by the CPU.	Continue to work without interruption.
DP stations to be removed:	as for I/O modules to be removed (see above)		

1) no longer contained in the hardware configuration, but still plugged in

2) Central modules are reset first in addition. Output modules briefly output 0 during this time (instead of the configured substitute or holding values).

Reaction if monitoring times are exceeded

If one of the monitored times exceeds the maximum value configured the update is interrupted and no change of master takes place. The fault-tolerant system remains in Solo mode with the previous master CPU and in certain conditions attempts to perform the change of master later. For more details refer to section 5.3.

10.3.6 PCS7, Step VI: Transition to the Redundant System Mode

Initial situation

The fault-tolerant system works with the new hardware configuration in Solo mode.

Procedure

1. In SIMATIC Manager select a CPU of the fault-tolerant system and select the menu command **PLC > Operating Mode**.
2. In the **Operating Mode** dialog box select the standby CPU and click the **Restart (warm restart)** button.

Result

Standby CPU links up again and is updated. The fault-tolerant system works with the new hardware configuration in the Redundant system mode.

Reaction of the I/O

Type of I/O	One-way I/O of standby CPU	One-way I/O of master CPU	Switched I/O
I/O modules to be removed[1)]	Are no longer addressed by the CPU. Driver blocks are no longer present.		
I/O modules that continue to be present	Are re-parameterized [2)] and updated by the CPU.	Continue to work without interruption.	
DP stations to be removed:	as for I/O modules to be removed (see above)		

1) no longer contained in the hardware configuration, but still plugged in

2) Central modules are first reset in addition. Output modules briefly output 0 during this time (instead of the substitute or holding values configured).

Reaction if monitoring times are exceeded

If one of the monitored times exceeds the maximum value configured the update is interrupted. The fault-tolerant system remains in Solo mode with the previous master CPU and in certain conditions attempts to perform the link-up and update later. For more details refer to Section 5.3.

10.3.7 PCS7, Step VII: Modification of Hardware

Initial situation

The fault-tolerant system works with the new hardware configuration in the Redundant system mode.

Procedure

1. Disconnect all the sensors and actuators from the components to be removed.
2. Unplug modules of the one-way I/O that are no longer required from the rack.
3. Unplug components that are no longer required from the modular DP stations.
4. Remove DP stations that are no longer required from the DP master systems.

Note

With switched I/O: End all modifications on **one** line of the redundant DP master system first before starting modifications to the second line.

Result

Unplugging modules that have been removed from the configuration has no effect on the user program. The same applies to the removal of DP stations.

The fault-tolerant system continues to work in the Redundant system mode.

10.4 Adding Components in STEP 7

Initial situation

You have ensured that the CPU parameters (e.g. the monitoring times) suit the planned new program. If necessary you must first change the CPU parameters accordingly (see Section 10.6).

The fault-tolerant system is working in the Redundant system mode.

Procedure

In order to add hardware components to a fault-tolerant system under STEP 7 the steps listed below are to be performed. Details of each step are listed in a subsection.

Step	What Has To Be Done?	Refer to Section
1	Modification of Hardware	10.4.1
2	Offline Modification of the Hardware Configuration	10.4.2
3	Expanding and Loading Organization Blocks	10.4.3
4	Stopping the Standby CPU	10.4.4
5	Loading new Hardware Configuration in the Standby CPU	10.4.5
6	Switch to CPU with modified configuration	10.4.6
7	Transition to the Redundant System Mode	10.4.7
8	Changing and Loading User Program	10.4.8

Exception

This overall process of system modification does not apply to the use of free channels on an existing module (see Section 10.4.9).

10.4.1 STEP 7, Step 1: Modification of Hardware

Initial situation

The fault-tolerant system is working in the Redundant system mode.

Procedure

1. Add the new components to the system.
   - Plug new central modules into the rack.
   - Plug new module into existing modular DP stations
   - Add new DP stations to existing DP master systems.

Note

With switched I/O: End all modifications on **one** line of the redundant DP master system first before starting modifications to the second line.

2. Connect the required sensors and actuators to the new components.

Result

Plugging in modules that have not yet been configured will have no effect on the user program. The same applies to adding DP stations.

The fault-tolerant system continues to work in the Redundant system mode.

New components are not yet addressed.

10.4.2 STEP 7, Step 2: Offline Modification of the Hardware Configuration

Initial situation

The fault-tolerant system is working in the Redundant system mode. The modules added will not yet be addressed.

Procedure

1. Perform all the modifications to the hardware configuration relating to the added hardware offline.
2. Compile the new hardware configuration, but do **not** load it into the PLC yet.

Result

The modified hardware configuration is in the PG. The PLC continues to work with the old configuration in the Redundant system mode.

Configuring connections

Connections from or to newly added CPs must be configured for both connection partners **after** modification of the hardware configuration is complete.

10.4.3 STEP 7, Step 3: Expanding and Loading Organization Blocks

Initial situation

The fault-tolerant system is working in the Redundant system mode.

Procedure

1. Make sure that the interrupt OBs 4x, 82, 83, 85, 86 and 122 react in the desired way to interrupts from the newly added components.
2. Load the modified OBs and the program segments affected by these into the PLC.

Result

The fault-tolerant system is working in the Redundant system mode.

10.4.4 STEP 7, Step 4: Stopping the Standby CPU

Initial situation

The fault-tolerant system is working in the Redundant system mode.

Procedure

1. In SIMATIC Manager select a CPU of the fault-tolerant system and select the menu command **PLC > Operating Mode**.
2. In the **Operating Mode** dialog box select the standby CPU and click the **Stop** button.

Result

The standby CPU switches to STOP mode, the master CPU remains in RUN mode, the fault-tolerant system works in Solo mode. One-way I/O of the standby CPU are no longer addressed. OB 70 (I/O redundancy loss) is not called due to the superior CPU redundancy loss (OB 72).

10.4.5 STEP 7, Step 5: Loading new Hardware Configuration in the Standby CPU

Initial situation

The fault-tolerant system is working in Solo mode.

Procedure

Load the compiled hardware configuration in the standby CPU that is in STOP mode.

Note

The user program and the connection configuration must not be overloaded in Solo mode.

Result

The new hardware configuration of the standby CPU does not yet have an effect on current operation.

10.4.6 STEP 7, Step 6: Switch to CPU with modified configuration

Initial situation

The modified hardware configuration is loaded into the standby CPU.

Procedure

1. In SIMATIC Manager select a CPU of the fault-tolerant system and select the menu command **PLC > Operating Mode**.
2. In the **Operating Mode** dialog box click the **Switch to button.**
3. In the **Switch to** dialog box select the option **with modified configuration** and click the **Switch** button.
4. Confirm the check question that follows with **OK**.

Result

The standby CPU connects, is updated and becomes the master. The former master CPU switches to STOP mode, the fault-tolerant system works with the new hardware configuration in Solo mode.

Behavior of the I/O

Type of I/O	One-way I/O of previous master CPU	One-way I/O of new master CPU	Switched I/O
Added I/O modules	Are not addressed by the CPU.	Are parameterized and updated by the CPU. Output modules briefly output the substitute values configured.	
I/O modules that continue to be present	Are no longer addressed by the CPU. Output modules output the configured substitute or holding values.	Are re-parameterized [1)] and updated by the CPU.	Continue to work without interruption.
Added DP stations	Are not addressed by the CPU.	as for added I/O modules (see above)	

1) Central modules are first reset in addition. Output modules briefly output 0 during this time (instead of the substitute or holding values configured).

Reaction if monitoring times are exceeded

If one of the monitored times exceeds the maximum value configured the update is interrupted and no change of master takes place. The fault-tolerant system remains in Solo mode with the previous master CPU and in certain conditions attempts to perform the change of master later. For more details refer to Section 5.3.

10.4.7 STEP 7, Step 7: Transition to the Redundant System Mode

Initial situation

The fault-tolerant system works with the new hardware configuration in Solo mode.

Procedure

1. In SIMATIC Manager select a CPU of the fault-tolerant system and select the menu command **PLC > Operating Mode**.
2. In the **Operating Mode** dialog box select the standby CPU and click the **Restart (warm restart)** button.

Result

Standby CPU links up again and is updated. The fault-tolerant system works with the new hardware configuration in the Redundant system mode.

Reaction of the I/O

Type of I/O	One-way I/O of standby CPU	One-way I/O of master CPU	Switched I/O
Added I/O modules	Are parameterized and updated by the CPU. Output modules briefly output the substitute values configured.	Are updated by the CPU.	Are updated by the CPU. Generate plug in interrupt; must be ignored in OB 83.
I/O modules that continue to be present	Are re-parameterized [1] and updated by the CPU.	Continue to work without interruption.	
Added DP stations	as for added I/O modules (see above)	Are updated by the CPU.	

1) Central modules are reset first in addition. Output modules briefly output 0 during this time (instead of the configured substitute or holding values).

Reaction if monitoring times are exceeded

If one of the monitored times exceeds the maximum value configured the update is interrupted. The fault-tolerant system remains in Solo mode with the previous master CPU and in certain conditions attempts to perform the link-up and update later. For more details refer to Section 5.3.

10.4.8 STEP 7, Step 8: Changing and Loading User Program

Initial situation

The fault-tolerant system works with the new hardware configuration in the Redundant system mode.

Restrictions

Caution

Structural modifications to an FB interface or the instance data of an FB are not possible in the Redundant system mode and result in the system mode Stop (both CPUs in STOP mode).

Procedure

1. Perform all the program modifications relating to the added hardware.

   You can add, modify or remove OBs, FBs, FCs and DBs.

2. Load only the program modifications into the PLC.
3. Configure the connections from or to the newly added CPs on both connection partners and load these into the PLC.

Result

The fault-tolerant system processes the entire system hardware with the new user program in the Redundant system mode.

10.4.9 Use of free channels on an existing module

The use of previously free channels of an I/O module depends primarily on whether or not parameters can be assigned to the module.

Modules to which parameters cannot be assigned

In the case of modules to which parameters cannot be assigned free channels can be connected and used in the user program at any time.

Modules to which parameters can be assigned

In the case of modules to which parameters can be assigned the hardware configuration must be modified to the sensors or actuators to be used. This generally necessitates reassignment of parameters to entire module.

It is then no longer possible to operate the module concerned without interruption.

- One-way output modules briefly output 0 (instead of the configured substitute or holding values).
- New parameters are not assigned to modules in switched DP stations on switching to the CPU with the modified configuration. To change the channel usage the following procedure is required:
  - In steps I to VI (see Section 10.5) the module concerned is completely removed from the hardware configuration and the user program. However, it can remain plugged into the DP station.
  - In steps 3 to 8 (see Section 10.4) the module is added to the hardware configuration and the user program again with the changed use.

Warning

Modules concerned will not be addressed between the two switching processes (steps V and 6); affected output modules output the value 0.

If this behavior cannot be tolerated for the process to be controlled then it is not possible to use previously free channels. In this case you must plug in additional modules in order to expand the system.

10.5 Removing Components in STEP 7

Initial situation

You have ensured that the CPU parameters (e.g. the monitoring times) suit the planned new program. If necessary you must first change the CPU parameters accordingly (see section 10.6).

The modules to be removed and the associated sensors and actuators are no longer of any significance for the process to be controlled. The fault-tolerant system is working in the Redundant system mode.

Procedure

In order to remove hardware components from a fault-tolerant system under STEP 7 the steps listed below are to be performed. Details of each step are listed in a subsection.

Step	What Has To Be Done?	Refer to Section
I	Offline Modification of the Hardware Configuration	10.5.1
II	Changing and Loading User Program	10.5.2
III	Stopping the Standby CPU	10.5.3
IV	Loading new Hardware Configuration in the Standby CPU	10.5.4
V	Switch to CPU with modified configuration	10.5.5
VI	Transition to the Redundant System Mode	10.5.6
VII	Modification of Hardware	10.5.7
VIII	Modifying and loading organization blocks	10.5.8

10.5.1 STEP 7, Step I: Offline Modification of the Hardware Configuration

Initial situation

The fault-tolerant system is working in the Redundant system mode.

Procedure

1. Perform offline all the modifications to the hardware configuration relating to the hardware to be removed.
2. Compile the new hardware configuration, but do **not** load it into the PLC yet.

Result

The modified hardware configuration is in the PG. The PLC continues to work with the old configuration in the Redundant system mode.

10.5.2 STEP 7, Step II: Changing and Loading User Program

Initial situation

The fault-tolerant system is working in the Redundant system mode.

Restrictions

Caution

Structural modifications to an FB interface or the instance data of an FB are not possible in the Redundant system mode and result in the system mode Stop (both CPUs in STOP mode).

Procedure

1. Perform only the program modifications relating to the hardware to be removed.

   You can add, modify or remove OBs, FBs, FCs and DBs.

2. Load only the program modifications into the PLC.

Result

The fault-tolerant system continues to work in the Redundant system mode. The modified user program will no longer attempt to access the hardware to be removed.

10.5.3 STEP 7, Step III: Stopping the Standby CPU

Initial situation

The fault-tolerant system is working in the Redundant system mode. The user program will no longer attempt to access the hardware to be removed.

Procedure

1. In SIMATIC Manager select a CPU of the fault-tolerant system and select the menu command **PLC > Operating Mode**.
2. In the **Operating Mode** dialog box select the standby CPU and click the **Stop** button.

Result

The standby CPU switches to STOP mode, the master CPU remains in RUN mode, the fault-tolerant system works in Solo mode. One-way I/O of the standby CPU are no longer addressed.

10.5.4 STEP 7, Step IV: Loading new Hardware Configuration in the Standby CPU

Initial situation

The fault-tolerant system is working in Solo mode.

Procedure

Load the compiled hardware configuration in the standby CPU that is in STOP mode.

Note

The user program and the connection configuration must not be overloaded in Solo mode.

Result

The new hardware configuration of the standby CPU does not yet have an effect on current operation.

10.5.5 STEP 7, Step V: Switch to CPU with modified configuration

Initial situation

The modified hardware configuration is loaded into the standby CPU.

Procedure

1. In SIMATIC Manager select a CPU of the fault-tolerant system and select the menu command **PLC > Operating Mode**.
2. In the **Operating Mode** dialog box click the **Switch to button.**
3. In the **Switch to** dialog box select the option **with modified configuration** and click the **Switch** button.
4. Confirm the check question that follows with **OK**.

Result

The standby CPU connects, is updated (see Chapter 5) and becomes the master. The former master CPU switches to STOP mode, the fault-tolerant system continues to work in Solo mode.

Behavior of the I/O

Type of I/O	One-way I/O of previous master CPU	One-way I/O of new master CPU	Switched I/O
I/O modules to be removed[1)]	Are no longer addressed by the CPU.		
I/O modules that continue to be present	Are no longer addressed by the CPU. Output modules output the configured substitute or holding values.	Are re-parameterized [2)] and updated by the CPU.	Continue to work without interruption.
DP stations to be removed:	as for I/O modules to be removed (see above)		

1) no longer contained in the hardware configuration, but still plugged in

2) Central modules are first reset in addition. Output modules briefly output 0 during this time (instead of the substitute or holding values configured).

Reaction if monitoring times are exceeded

If one of the monitored times exceeds the maximum value configured the update is interrupted and no change of master takes place. The fault-tolerant system remains in Solo mode with the previous master CPU and in certain conditions attempts to perform the change of master later. For more details refer to Section 5.3.

10.5.6 STEP 7, Step VI: Transition to the Redundant System Mode

Initial situation

The fault-tolerant system works with the new (restricted) hardware configuration in Solo mode.

Procedure

1. In SIMATIC Manager select a CPU of the fault-tolerant system and select the menu command **PLC > Operating Mode**.
2. In the **Operating Mode** dialog box select the standby CPU and click the **Restart (warm restart)** button.

Result

Standby CPU links up again and is updated. The fault-tolerant system is working in the Redundant system mode.

Behavior of the I/O

Type of I/O	One-way I/O of standby CPU	One-way I/O of master CPU	Switched I/O
I/O modules to be removed[1)]	Are no longer addressed by the CPU.		
I/O modules that continue to be present	Are re-parameterized [2)] and updated by the CPU.	Continue to work without interruption.	
DP stations to be removed:	as for I/O modules to be removed (see above)		

1) no longer contained in the hardware configuration, but still plugged in

2) Central modules are first reset in addition. Output modules briefly output 0 during this time (instead of the substitute or holding values configured).

Behavior if monitoring times are exceeded

If one of the monitored times exceeds the maximum value configured the update is interrupted. The fault-tolerant system remains in Solo mode with the previous master CPU and in certain conditions attempts to perform the link-up and update later. For more details refer to Section 5.3.

10.5.7 STEP 7, Step VII: Modification of Hardware

Initial situation

The fault-tolerant system works with the new hardware configuration in the Redundant system mode.

Procedure

1. Disconnect all the sensors and actuators from the components to be removed.
2. Remove the desired components from the system.
   - Unplug central modules from the rack.
   - Unplug modules from modular DP stations
   - Remove DP stations from DP master systems.

Note

With switched I/O: End all modifications on **one** line of the redundant DP master system first before starting modifications to the second line.

Result

Unplugging modules that have been removed from the configuration has no effect on the user program. The same applies to the removal of DP stations.

The fault-tolerant system continues to work in the Redundant system mode.

10.5.8 STEP 7, Step VIII: Modifying and loading organization blocks

Initial situation

The fault-tolerant system is working in the Redundant system mode.

Procedure

1. Make sure that the interrupt OBs 4x and 82 no longer react to interrupts from the removed components.
2. Load the modified OBs and the program segments affected by these into the PLC.

Result

The fault-tolerant system is working in the Redundant system mode.

10.6 Changing the CPU Parameters

Only certain parameters (object properties) of the CPUs can be modified while in operation. They are identified in the screen form by blue text (if you have set blue as the color for dialog box text on the Windows Control Panel, the modifiable parameters are shown in black).

Note

If you modify parameters that should not be changed, there is no automatic transfer to the CPU whose parameters have been modified. In this case the event W#16#5966 is entered in the diagnostic buffer.

Table 10-1 Modifiable CPU Parameters

Tab	Modifiable Parameter
Startup	Monitoring time for signaling readiness by modules
	Monitoring time for transferring parameters to modules
Scan cycle/clock memory	Scan cycle monitoring time
	Cycle load due to communications
	Size of the process image of inputs
	Size of the process image of outputs
Memory	Local data (for the different priority classes)
	Communication resources: maximum number of communication jobs (you are only allowed to increase this parameter compared to its previously configured value.)
Time-of-day interrupts (for every time-of-day interrupt OB)	“Active” check box
	“Execution” list box
	Starting date
	Time
Watchdog interrupt (for every watchdog interrupt OB)	Execution
	Phase offset
Diagnostics/clock	Correction factor
Security	Protection level and password
H parameters	Test scan cycle time
	Maximum scan-cycle time extension
	Maximum communication delay
	Maximum hold time for priority classes > 15
	Minimum I/O hold time

The new values are to be chosen to suit both the user program currently loaded and the new user program planned.

Initial situation

The fault-tolerant system is working in the Redundant system mode.

Procedure

To change the CPU parameters of a fault-tolerant system the steps listed below are to be performed. Details of each step are listed in a subsection.

Step	What Has To Be Done?	Refer to Section
A	Changing the CPU Parameters Offline	10.6.1
B	Stopping the Standby CPU	10.6.2
C	Loading modified CPU parameters in the standby CPU	10.6.3
D	Switch to CPU with modified configuration	10.6.4
E	Transition to the Redundant System Mode	10.6.5

10.6.1 Step A: Changing the CPU Parameters Offline

Initial situation

The fault-tolerant system is working in the Redundant system mode.

Procedure

1. Change the desired properties of the CPU offline in the hardware configuration.
2. Compile the new hardware configuration, but do **not** load it into the PLC yet.

Result

The modified hardware configuration is in the PG/ES. The PLC continues to work with the old configuration in the Redundant system mode.

10.6.2 Step B: Stopping the Standby CPU

Initial situation

The fault-tolerant system is working in the Redundant system mode.

Procedure

1. In SIMATIC Manager select a CPU of the fault-tolerant system and select the menu command **PLC > Operating Mode**.
2. In the **Operating Mode** dialog box select the standby CPU and click the **Stop** button.

Result

The standby CPU switches to STOP mode, the master CPU remains in RUN mode, the fault-tolerant system works in Solo mode. One-way I/O of the standby CPU are no longer addressed.

10.6.3 Step C: Loading new Hardware Configuration in the Standby CPU

Initial situation

The fault-tolerant system is working in Solo mode.

Procedure

Load the compiled hardware configuration in the standby CPU that is in STOP mode.

Note

The user program and the connection configuration must not be overloaded in Solo mode.

Result

The modified CPU parameters in the new hardware configuration of the standby CPU do not yet have an effect on ongoing operation.

10.6.4 Step D: Switch to CPU with modified configuration

Initial situation

The modified hardware configuration is loaded into the standby CPU.

Procedure

1. In SIMATIC Manager select a CPU of the fault-tolerant system and select the menu command **PLC > Operating Mode**.
2. In the **Operating Mode** dialog box click the **Switch to button.**
3. In the **Switch to** dialog box select the option **with modified configuration** and click the **Switch** button.
4. Confirm the check question that follows with **OK**.

Result

The standby CPU connects, is updated and becomes the master. The former master CPU switches to STOP mode, the fault-tolerant system continues to work in Solo mode.

Behavior of the I/O

Type of I/O	One-way I/O of previous master CPU	One-way I/O of new master CPU	Switched I/O
I/O modules	Are no longer addressed by the CPU. Output modules output the configured substitute or holding values.	Are re-parameterized [1] and updated by the CPU.	Continue to work without interruption.

1) Central modules are first reset in addition. Output modules briefly output 0 during this time (instead of the substitute or holding values configured).

Behavior if monitoring times are exceeded

If one of the monitored times exceeds the maximum value configured the update is interrupted and no change of master takes place. The fault-tolerant system remains in Solo mode with the previous master CPU and in certain conditions attempts to perform the change of master later. For more details refer to Section 5.3.

If the values for the monitoring times in the CPUs are different then the higher value always applies.

10.6.5 Step E: Transition to the Redundant System Mode

Initial situation

The fault-tolerant system works with the modified CPU parameters in Solo mode.

Procedure

1. In SIMATIC Manager select a CPU of the fault-tolerant system and select the menu command **PLC > Operating Mode**.
2. In the **Operating Mode** dialog box select the standby CPU and click the **Restart (warm restart)** button.

Result

Standby CPU links up again and is updated. The fault-tolerant system is working in the Redundant system mode.

Reaction of the I/O

Type of I/O	One-way I/O of standby CPU	One-way I/O of master CPU	Switched I/O
I/O modules	Are re-parameterized [1] and updated by the CPU.	Continue to work without interruption.	

1) Central modules are reset first in addition. Output modules briefly output 0 during this time (instead of the configured substitute or holding values).

Reaction if monitoring times are exceeded

If one of the monitored times exceeds the maximum value configured the update is interrupted. The fault-tolerant system remains in Solo mode with the previous master CPU and in certain conditions attempts to perform the link-up and update later. For more details refer to Section 5.3.

If the values for the monitoring times in the CPUs are different then the higher value always applies.

10.7 Changing the Memory Components of the CPU

The Redundant system mode is only possible if the two CPUs have the same memory components. For this, the following conditions must be met:

- The main memory of the two CPUs must be the same size.
- The load memory of the two CPUs must be the same size and of the same type (RAM or FLASH).

The memory components of the CPUs can be modified while in operation. Possible memory modifications in S7-400H are:

- expansion of the main memory and/or load memory
- changing the type of load memory

10.7.1 Expand the main and/or load memory

The following methods of memory expansion are possible:

- expanding the main memory by plugging in additional or larger memory modules
- expanding the load memory by plugging in a larger memory card of the same type instead of the existing one
- expanding the load memory by plugging in a RAM card if no memory card was plugged in previously

With this type of memory change the complete user program is copied from the master CPU to the standby CPU on link-up (see section 5.2.1).

Restrictions

Expanding the load memory is only meaningful in the case of RAM Cards, since only then can the user program be copied to the load memory of the standby CPU on link-up.

In principle it is also possible to expand the load memory in the form of FLASH cards, but it is then the user's responsibility to load the complete user program and the hardware configuration into the new FLASH card (see procedure in Section 10.7.2).

Initial situation

The fault-tolerant system is working in the Redundant system mode.

Procedure

Perform the steps below in the order specified:

Step	What Has To Be Done?	How Does the System React?
1	Switch the standby CPU to STOP mode using the PG.	The system is working in Solo mode.
2	If you want to expand the main memory: A Turn off the power supply for the standby CPU. B Unplug the standby CPU from the central controller (CC). C Install the desired memory modules as described in the installation manual *S7-400, M7-400 Programmable Controllers, Hardware and Installation*. D Plug the CPU back into in the CC. E Make sure that the synchronization link is re-established and that the Operating Mode switch of the standby CPU is switched to RUN or RUN-P. F Turn the power supply for the standby CPU back on.	Subsystem is disabled for periods.
3	If you want to expand the load memory: Unplug the existing memory card from the CPU and plug in a memory card of the same type with the desired (larger) capacity.	Standby CPU requests reset.
4	Reset the standby CPU using the PG.	–
5	Start the standby CPU by means of the menu command **PLC > Mode > Switch to CPU with... expanded memory configuration.**	• Standby CPU connects, is updated and becomes the master. • Previous master CPU goes to STOP. • System works in Solo mode.
6	Turn off the power supply for the second CPU.	Subsystem is disabled.
7	Modify the memory components of the second CPU as you did for the first CPU in steps 2 to 4.	–
8	Start the second CPU using the PG.	• Second CPU links up and is updated. • System works the Redundant system mode again.

10.7.2 Changing the type of load memory

The following types of memory cards are available as load memory:

- RAM card for the test and commissioning phase
- FLASH card for the permanent storage of the finished user program

The size of the new memory card is irrelevant here.

With this type of memory modification no program segments are transferred from the master CPU to the standby CPU, only the contents of the blocks in the user program that remain unchanged (see section 5.2.3).

It is the user's responsibility to load the entire user program into the new load memory.

Initial situation

The fault-tolerant system is working in the Redundant system mode.

The current status of the user program is available on the PG/ES as a STEP 7 project in modular form.

Caution

You cannot use a user program loaded from the PLC here.

It is not permissible to recompile the user program from an STL source file since then all the blocks will be given a new time stamp. No block contents will then be copied on master/standby switch-over.

Procedure

Perform the steps below in the order specified:

Step	What Has To Be Done?	How Does the System React?
1	Switch the standby CPU to STOP mode using the PG.	The system is working in Solo mode.
2	Unplug the existing memory card from the standby CPU and plug in a memory card of the desired type.	Standby CPU requests reset.
3	Reset the standby CPU using the PG.	–
4	Load the user program and the hardware configuration into the standby CPU.	–
5	Start the standby CPU by means of the menu command **PLC > Mode > Switch to CPU with... modified configuration.**	• Standby CPU connects, is updated and becomes the master. • Previous master CPU goes to STOP. • System works in Solo mode.

Step	What Has To Be Done?	How Does the System React?
6	Modify the memory components of the second CPU as you did for the first CPU in step 2.	–
7	Load the user program and the hardware configuration into the second CPU.	–
8	Start the second CPU using the PG.	• Second CPU links up and is updated. • System works the Redundant system mode again.

Note

If you want to change to FLASH cards you can load these outside the CPU with the user program and hardware configuration. You can then omit steps 4 and 7.

However, the memory cards in the two CPUs must be loaded using the same procedure. Changing the order of the blocks in the load memories will result in cancellation of the link-up.

10.8 Perform operating system update

The Redundant system mode is possible if both CPUs are working with the same operating system version. An operating system update can be performed while in operation.

Note

An operating system update is allowed only with certain specific firmware versions to the next higher firmware version.

Procedure

To modify the operating system during operation, perform the following steps:

Step	What Has To Be Done?	How Does the System React?
1	Switch the standby CPU to STOP mode using the PG.	The system is working in Solo mode.
2	Turn off the power supply for the standby CPU.	Subsystem is disabled.
3	Unplug the existing memory card from the standby CPU and plug in the memory card with the new operating system.	–
4	Turn the power supply for the standby CPU back on.	The standby CPU loads the new operating system and performs a reset.
5	Turn off the power supply for the standby CPU again.	Subsystem is disabled.
6	Unplug the memory card with the operating system from the standby CPU and plug in the original one again.	–
7	Turn the power supply for the standby CPU back on.	–
8	Start the standby CPU by means of the menu command **PLC > Mode > Switch to CPU with... modified operating system.**	• Standby CPU connects, is updated and becomes the master (all blocks are transferred). • Previous master CPU goes to STOP. • System works in Solo mode.
9	Perform steps 2 to 7 for the second CPU (previous master CPU).	– see above –
10	Start the second CPU using the PG.	• Second CPU connects and is updated (all blocks are transferred). • System works the Redundant system mode again.

Characteristic Values of Redundant Programmable Logic Controllers

A

The present appendix presents a brief introduction to the characteristic values of redundant programmable logic controllers and shows the practical effects of redundant configuration types by means of a few selected configurations.

A.1 Basic Concepts

The parameters normally used for a quantitative assessment of redundant programmable logic controllers are reliability and availability, which are described in further detail below.

Reliability

Reliability is the characteristic of a technical device to fulfill its function during its operating period. This is usually no longer possible when a component fails.

The criterion frequently specified for reliability is therefore the **MTBF** (**M**ean **T**ime **B**etween **F**ailures). It can be determined statistically by systems that are operating or by calculating the failure rates of the components used.

Reliability of modules

The reliability of SIMATIC components is extremely high as a consequence of wide-ranging quality assurance measures in development and manufacture.

The following average values apply to SIMATIC modules:

- MTBF of a central processing unit: 15 years
- MTBF of an I/O module: 50 years

Reliability of programmable logic controllers

The use of redundant modules prolongs the MTBF of a system to a very large extent. In connection with the high-quality self-tests and the mechanisms for error detection, which are integrated in the CPUs of the S7-400H, virtually all errors are discovered and localized. The diagnostic coverage (dc) is approximately 95 percent.

Starting from the reliability of a single system (1-out-of-1 systems having an $MTBF_{1oo1}$), it is possible to calculate the reliability of the S7-400H as a two-channel (1-out-of-2) fault-tolerant system from the following formula:

$$MTBF_{1oo2} = \frac{MTBF_{1oo1}^{2}}{2MDT + 2(1 - dc) \cdot MTBF_{1oo1}}$$

The MTBF of the S7-400H is determined by its **MDT** (**M**ean **D**own **T**ime). This time consists essentially of the time for error detection and the time required to repair or replace defective modules.

The error detection time is half the configured test cycle time (by default, 90 min.). The repair time for a modular system such as the S7-400H is normally four hours.

Availability

Availability is the probability of a system being capable of operation at a specified point of time. It can be enhanced by means of redundancy – for example, by using redundant I/O modules or by using multiple sensors at one sampling point. Redundant components are arranged such that system operability is not affected by the failure of a single component. Here, again, an important element of availability is a detailed diagnostic display.

The availability of a system is expressed as a percentage. It is determined by the mean time between failure (MTBF) and the mean repair time (MTTR). The availability of a two-channel (1-out-of-2) fault-tolerant system can be calculated from the following formula:

$$V = \frac{MTBF_{1oo2}}{MTBF_{1oo2} + MDT} 100\%$$

A.2 Comparison of MTBFs for Selected Configurations

The following sections compare systems having a central I/O.

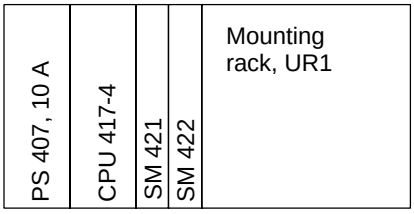
The following framework conditions are set for the calculation.

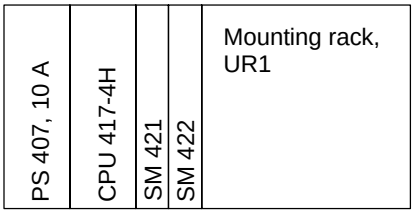
- MDT (Mean Down Time) 4 hours
- Ambient temperature 40 degrees
- Buffer voltage is guaranteed

A.2.1 System Configurations With Central I/O

The following system with a standard CPU 417-4 is taken as a basis for calculating a reference factor which specifies the multiple of the availability of the other systems having a central I/O compared with the baseline.

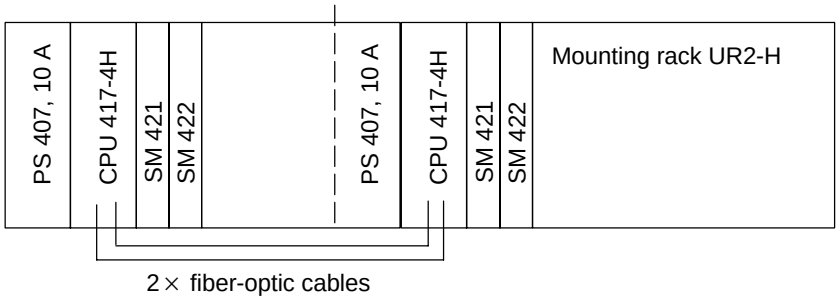
Standard and fault-tolerant CPU in single operation

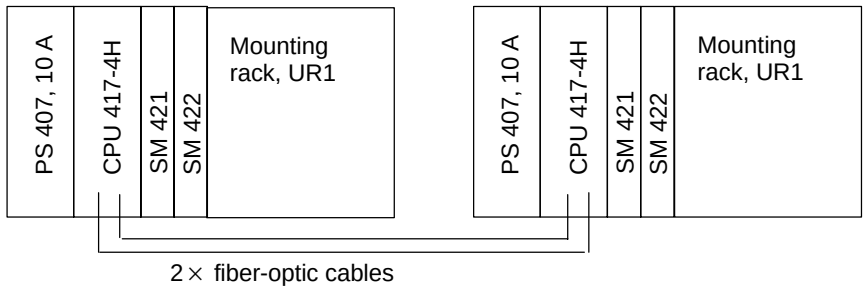
Standard CPU 417-4	Baseline
	1

CPU with high availability 417-4H in Solo mode *)	Factor
	1

*) not possible with firmware version V2.0.0

Redundant CPUs in different mounting racks

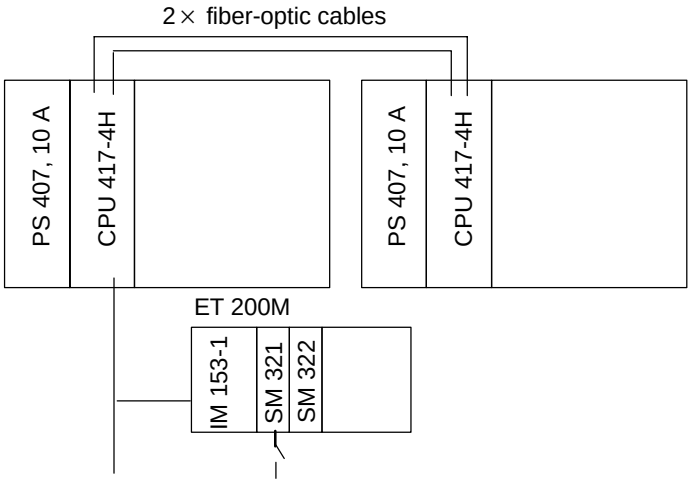
Redundant CPU 417-4 H in split mounting rack	Factor
	23

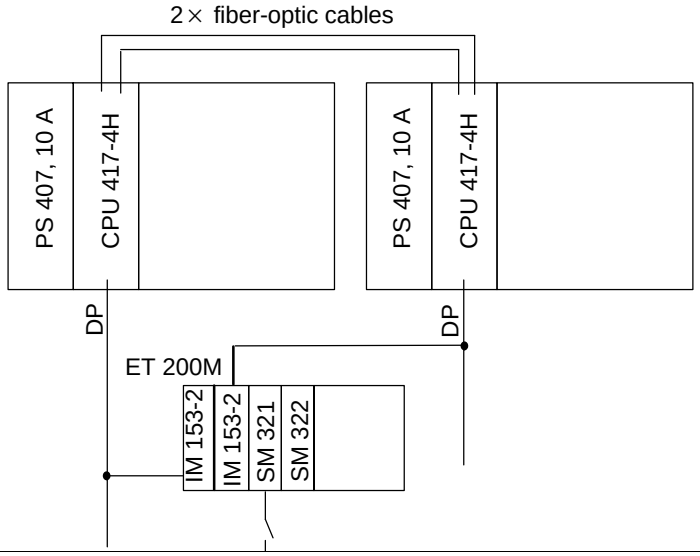
Redundant CPU 417-4H in separate mounting racks	Factor
	59

A.2.2 System Configurations With Distributed I/O

The following system with two fault-tolerant CPUs 417-4 H and a one-sided I/O is taken as a basis for calculating a reference factor which specifies the multiple of the availability of the other systems with a distributed I/O compared with the baseline.

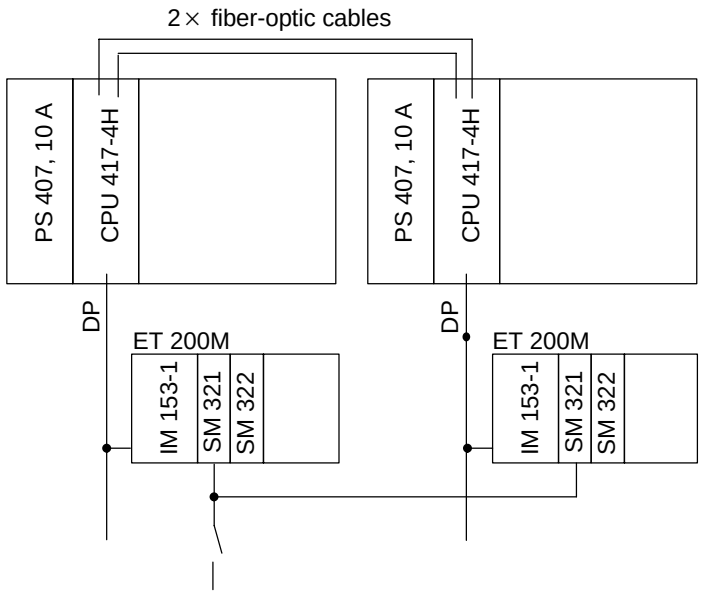
Redundant CPUs with a single-channel, one-sided or switched I/O

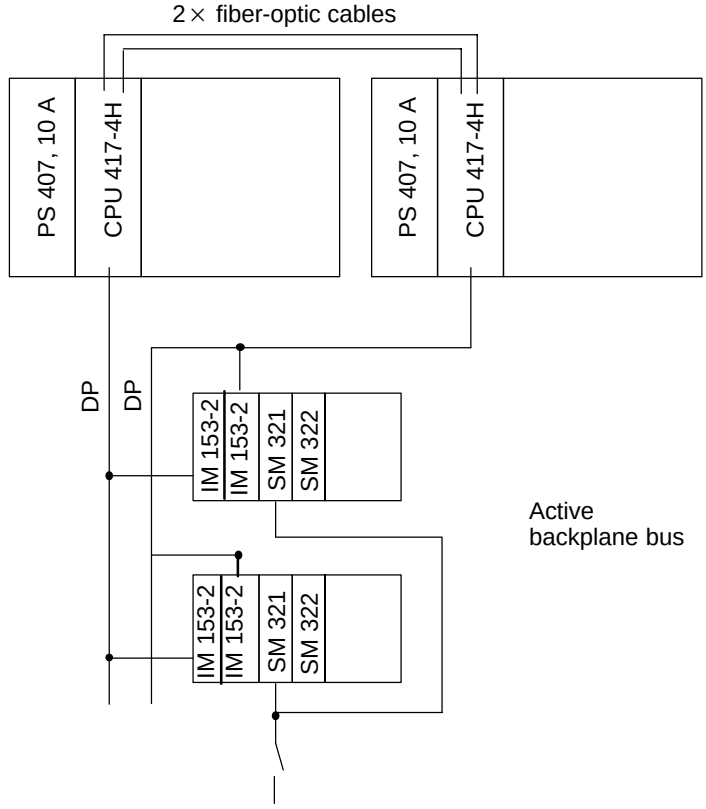
One-sided, distributed I/O	Baseline
	1

Switched distributed I/O	Factor
	3 or 12 *

* A factor of 3 applies to the condition when the failure of an I/O module has resulted in the entire system stopping. A factor of 12 applies when the failure of an I/O module has not resulted in the entire system stopping.

Redundant CPUs with redundant I/O

Redundant one-sided I/O	Factor
	65

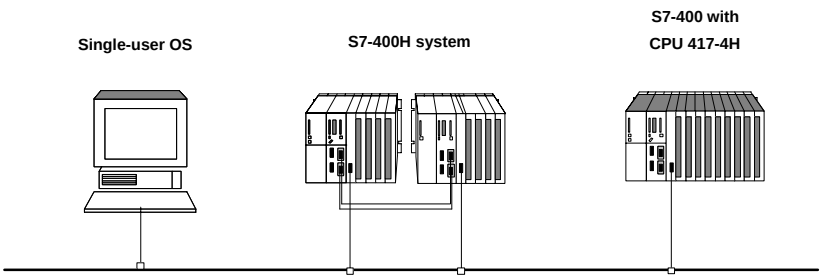
Redundant switched I/O	Factor
	70

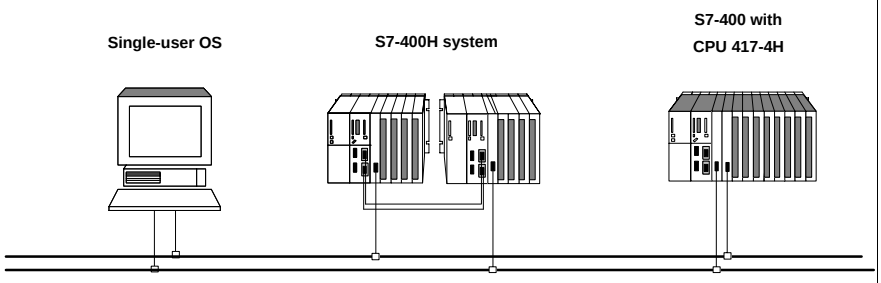
A.2.3 Comparison of System Configurations With Standard and Fault-Tolerant Communications

The following section shows a comparison between standard and fault-tolerant communications for a configuration consisting of a fault-tolerant system, a fault-tolerant CPU 417-4 H and a single-channel OS.

By comparison, only the communication components CP and cable were taken into account.

Systems having standard and fault-tolerant communications

Standard communications	Baseline
	1

Fault-tolerant communications	Factor
	83

Separate Operation

B

Overview

The present appendix provides the information necessary for separate operation of a CPU 417-4H. You will learn in the following

- how separate operation is defined
- when separate operation is necessary
- what you have to take into account with separate operation
- how the H-specific LEDs respond
- how you configure a CPU 417-4H for separate operation
- how you can expand it to form a fault-tolerant system

The differences compared to a standard S7-400 CPU that you have to take into account when configuring and programming the CPU 417-4H are contained in Appendix D.

Definition

By separate operation, we understand use of a CPU 417-4H in a standard SIMATIC-400 station.

Reasons for separate operation

The following applications are only possible with a CPU 417-4H – in other words, they are not possible with standard CPUs in the S7-400 range.

- Use of fault-tolerant connections
- Configuration of the S7-400F fail-safe programmable logic controller

  A fail-safe user program can only be compiled as an executable when using the CPU 417-4H with a fail-safe runtime license (for more information refer to the manual *Programmable Logic Controllers S7-400F and S7-400FH Fail-Safe Systems*.).

Note

The CPU 417-4H self-test can also be performed in separate operation.

What you have to take into account with the separate operation of a CPU 417-4H

Note

Synchronization submodules must not be inserted when a CPU 417-4H is operated separately.

Compared to a standard S7-400 CPU, a CPU 417-4H features additional functions, but is does not support certain functions. You must therefore know on which CPU your user program should run, especially when you are programming your programmable logic controller. A user program that you have written for a standard S7-400 CPU will therefore not normally run on a CPU 417-4H in separate operation without adjustments.

The following table lists the differences between operation of a standard S7-400 CPU and separate operation of a CPU 417-4H.

Function	Standard S7-400 CPU	CPU 417-4H in Separate Operation
Use of symbol-oriented messages (SCAN)	Yes	No
Multicomputing (OB60, SFC35)	Yes	No
Startup without configuration loaded	Yes, if no IMs, no CPs and FMs are plugged in and no expansion units are connected	No
Insertion of DP modules in the module slots for interface modules	Yes	No. The module slots are only intended for the synchronization submodules.
Connection of S5 modules via IM or adapter casings	Yes	No
Self-test after POWER ON	No	Yes
Redundancy error OBs (OB70, OB72)	No	Yes, but no calls
Preset priority class for OBs 81 to 87	26	25
Response of the CPU upon overflow of the start information buffer in priority class 26	STOP (event W#16#4541)	Call OB80 in priority class 28. If OB not loaded, STOP.
Response of the CPU upon overflow of the startup information buffer in priority class 28	This event cannot occur.	Entry in diagnostic buffer
Background processing (OB90)	Yes	No
Restart (OB101)	Yes	No

Function	Standard S7-400 CPU	CPU 417-4H in Separate Operation
Specify the rack number and the CPU in the OB start information	No	Yes
SSL ID W#16#0019 (status of all LEDs)	No data records for the H-specific LEDs	Data records for all LEDs
SSL ID W#16#0222 (data record for the specified interrupt)	No data record for the redundancy error OBs (OB70, OB72)	Data records for all interrupt OBs
SSL ID W#16#0232 index W#16#0004 byte 0 of the word “index” in the data record	W#16#00	W#16#F8
SSL ID W#16#xy71 fault-tolerant CPU group information	No	Yes
SSL ID W#16#0174 (status of a module LED)	No data record for the H-specific LEDs	Data records for all LEDs
Specify the rack number and the CPU in diagnostic buffer entries	No	Yes
Direct communication between DP slaves	Yes	Cannot be configured in STEP 7
Equidistance of DP slaves	Yes	No
Groups of DP slaves synchronize with SFC11 “DPSYC_FR”	Yes	No
Global data communication	Yes	No: neither cyclically, nor by means of SFC60 “GD_SND” and SFC61 “GD_RCV”
S7 basic communication	Yes	No
SFC90 “H_CTRL”	No	Yes

H-specific LEDs

LEDs REDF, IFM1F, IFM2F, MSTR, RACK0 and RACK1 show the response specified in the following table.

LED	Response
REDF	Dark
IFM1F	Dark
IFM2F	Dark
MSTR	Lights
RACK0	Lights
RACK1	Dark

Configuring separate operation

Requirement: The “S7 Fault-Tolerant Systems” option pack must be installed.

Perform the following steps:

1. Insert a SIMATIC-400 station in your project.
2. Configure the station with the CPU 417-4H in accordance with your hardware configuration. For separate operation, you must insert the CPU 417-4H in a standard rack (Insert > Station > S7-400 Station in SIMATIC Manager).
3. Assign parameters to the CPU 417-4H. You can use the default values or adapt the necessary parameters.
4. Configure the necessary networks and connections. For separate operation, you can configure “fault-tolerant S7 connection” type connections.

You will find help on the procedure in the SIMATIC Manager Help topics and in the Help of the “S7 Fault-Tolerant Systems” option pack.

Upgrading to a fault-tolerant system

Note

Upgrading to a fault-tolerant system is only possible if you have not assigned any odd numbers for expansion units in separate operation.

If you want to upgrade the CPU 417-4 H later to a fault-tolerant system, perform the following steps:

1. Open a new project and insert an H station.
2. Copy the complete rack from the standard SIMATIC-400 station and insert it twice in the H station.
3. Insert the necessary subnets.
4. If required, copy the DP slaves from the old separate operation project in the H station.
5. Reconfigure the communication links.
6. Perform modifications as necessary – by inserting one-sided I/O.

The steps you have to take during configuring are described in the online Help of the “S7-400H” option pack.

Installing and starting the fault-tolerant system

We recommend you to perform the following steps when installing and starting up the fault-tolerant system.

1. Save the configured fault-tolerant system to a flash memory card.
2. When installing the synchronization submodules, make sure the rack numbers are set correctly.
3. Do not connect the synchronization submodules with fiber-optic cables.
4. Plug the flash memory card into the central processing unit concerned and start it. Then the Stop LED flashes (reset request).
5. Perform a manual reset for both central processing units.
6. Connect the two central processing units with fiber-optic cable.
7. Start the two central processing units.

The fault-tolerant system then works in Redundant system mode.

Converting from S5-H to S7-400H

C

This appendix will help you to convert to fault-tolerant S7 systems if you are already familiar with fault-tolerant systems of the S5 family.

Generally speaking, knowledge of the STEP 7 configuration software is required for converting from the S5-H to the S7-400H.

C.1 General Information

Documentation

The following manuals are available for familiarization with the STEP 7 standard software:

- *Configuring Hardware and Communication Connections STEP 7 V5.1*
- *Programming with STEP 7 V5.1*

The following reference manuals describe the individual programming languages.

- *System and Standard Functions*
- *STL, LAD, FBD for S7-300/400*

The *From S5 to S7* manual will support you while you are converting and provides detailed information.

C.2 Configuration, Programming and Diagnostics

Configuration

In STEP5, configuration was performed with a separate configuration package – for example, COM 155H.

In STEP 7 we use the standard software in conjunction with the option package “S7 H Systems” to configure the fault-tolerant CPUs. Using SIMATIC Manager, create an H station and configure it with HWCONFIG. The special features of the fault-tolerant CPUs are summarized in a few registers. The integration in networks and the configuration of connections are performed with NetPro.

Diagnostics and programming

Error diagnostics on the S5 is implemented with the help of the error data block, into which the system enters all errors. The error OB 37 is started automatically for every entry. Further information has been stored in the H memory word.

The H memory word consists of a status byte and a control byte. Control information can be set bit by bit in the STEP5 user program.

In STEP 7, system diagnostics is accomplished by means of the diagnostic buffer or by displaying what are known as partial lists from the system status list (specific information for fault-tolerant systems, for example, are located in SSL71). This check can be performed with the help of the programming device or by the user program with SFC 51 “RDSYSST”.

OB 70 and OB 72 are available for I/O redundancy loss and CPU redundancy loss, respectively.

The function of the control byte is implemented in STEP 7 by means of the SFC 90 H_CTRL.

Subject with S5	Equivalent in S7
Error OB37	Error OBs OB 70 and OB 72
Memory control word	SFC H_CTRL 90
Memory status word	SSL71
Error block	Diagnostic buffer

Differences between Fault-Tolerant Systems and Standard Systems

D

Certain differences from standard S7-400 CPUs have to be taken into account when you configure and program a fault-tolerant programmable logic controller containing a CPU 417-4H. On the one hand, compared to a standard S7-400 CPU, a CPU 417-4H has additional functions, while on the other hand a CPU 417-4 does not support certain functions. This has to be taken in account in particular when you wish to run a program that was created for a standard S7-400 CPU on a CPU417-4H.

The items in which the programming of fault-tolerant systems differs from that for standard systems are summarized below.

If you use one of the calls concerned (OBs and SFCs) in your user program, you will need to adapt your program accordingly.

Additional functions of fault-systems

Function	Additional Programming
Redundancy error OBs	• I/O redundancy error OB (OB 70) • CPU redundancy error OB (OB 72) You will find detailed information in the Reference Manual *System and Standard Functions*
Additional information in OB start information and in diagnostic buffer entries	The rack number and the CPU (master/standby) are specified. You can evaluate this additional information in the program.
SFC for fault-tolerant systems	You can use SFC 90 “H_CTRL” to influence the sequences of fault-tolerant systems.
Fault-tolerant communication connections	Fault-tolerant connections are configured, no further programming is required. You can use the SFBs for configured connections when you are using fault-tolerant connections.
Self-test	The self-test is performed automatically, no further programming is required,
Switched I/O	No additional programming is required, refer to Section 6.3

Function	Additional Programming
Information on the system status list	• You also obtain data records for the H-specific LEDs by means of the partial list with the SSL ID W#16#0019. • You also obtain data records for the redundancy error OBs by means of the partial list with the SSL ID W#16#0222. • You obtain information on the current state of the fault-tolerant system by means of the partial list with SSL ID W#16#xy71. • You also obtain data records for the H-specific LEDs by means of the partial list with the SSL ID W#16#0174. • The partial list with the SSL-ID W#16#xy75 provides you with information on the status of the communications between the fault-tolerant system and switched DP slaves.
Monitoring during update	The operating system monitors the following four configurable timers: • maximum scan-cycle time extension • maximum communication delay • maximum hold time for priority classes > 15 • minimum I/O hold time No additional programming is required for this. For more details refer to Chapter 5.

Constraints with CPU 417-4 H compared to CPU 417-4

Function	Constraint with CPU 417-4 H
Use of symbol-oriented messages (SCAN)	Use of symbol-oriented messages is not possible.
Warm Restart	A warm restart is not possible. OB101 is not supported.
Multicomputing	Multicomputing is not possible. OB60 and SFC35 are not supported,
Startup without configuration loaded	Startup without loaded configuration is not possible.
Background OB	OB 90 is not supported.
CPU hardware error	OB 84 is not supported. Should a sporadic interface error occur the CPU enters the error in the diagnostic buffer and continues running.
Global data communication	GD communication is not possible (neither periodically nor by calling system functions SFC60 “GD_SND” and SFC61 “GD_RCV”)
Basic communication	Communication functions (system functions) for basic communication are not supported.

Function	Constraint with CPU 417-4 H
Direct communication between DP slaves	Cannot be configured in STEP 7
Equidistance of DP slaves	No equidistance for DP slaves in the fault-tolerant system
Synchronizing DP slaves	The synchronization of DP slave groups is not possible. SFC 11 “DPSYC_FR” is not supported.
Non-initialized local data	When local data are stored in a data area (memory markers, data blocks, etc.) or if they influence program execution, the local data have to be initialized. Non-initialized local data result in a synchronization error on a fault-tolerant system. The system goes to Solo mode with one CPU at STOP.
Runtime response	The command execution time is slightly slower on the CPU 417-4H than on the CPU 417-4 (refer to *Operations list S7-400*). This is to be taken into account in all time-critical applications. You may need to increase the scan cycle monitoring time.
Delays and blocks	During update: • the asynchronous SFCs for data records are given a negative acknowledgement • messages are delayed • all priority classes up to 15 are initially delayed • communications jobs are refused or delayed • finally all priority classes are blocked. For more details refer to Chapter 5.

Function Modules and Communication Processors Used on the S7-400H

E

You can use the following function modules (FMs) and communication processors (CPs) on a S7-400H programmable logic controller:

FMs for central use

Caution

At present you cannot use FMs centrally with the S7-400H.

CPs for central use

Module	Order no.	Release	one-way	redun-dant
Communications processor CP441-1 (point-to-point connection)	6ES7441-1AA02-0AE0	hardware version 2 or later	Yes	No
	6ES7441-1AA03-0XE0	hardware version 1 or later with firmware version V1.0.0 or higher		
Communications processor CP441-2 (point-to-point connection)	6ES7441-2AA02-0AE0	hardware version 2 or later	Yes	No
	6ES7441-2AA03-0XE0	hardware version 1 or later with firmware version V1.0.0 or higher		
Communications processor CP443-1 (Industrial Ethernet, ISO transport) 1)	6GK7443-1BX01-0XE0	hardware version 1 or later with firmware version V1.0.0 or higher	Yes	Yes
Communications processor CP443-1 TCP (Industrial Ethernet, TCP transport) 1)	6GK7443-1EX01-0XE0	hardware version 1 or later	Yes	No
Communications processor CP443-1 Multi (Industrial Ethernet, TCP/ISO transport)	6GK7443-1EX02-0XE0	hardware version 1 or later with firmware version V5.0.0 or higher	Yes	Yes
Communications processor CP443-1 Multi (Industrial Ethernet, TCP/ISO transport)	6GK7443-1EX10-0XE0	hardware version 1 or later with firmware version V1.0.0 or higher	Yes	Yes

Module	Order no.	Release	one-way	redun-dant
Communications processor CP342-2 (ASI bus interface module)	6GK7342-2AH01-0XB0	hardware version 1 or later with firmware version V1.10 or higher	Yes	Yes
Communications processor CP443-5 Basic (PROFIBUS; S7 communications)	6GK7443-5FX01-0XE0	hardware version 1 or later	Yes	Yes
Communications processor CP443-5 Extended (PROFIBUS; master on the DP PROFIBUS)	6GK7443-5DX02-0XE0	hardware version 1 or later with firmware version V3.0.0 or higher	Yes	Yes

[1)] has to be upgraded to MLFB 6GK7 443-1EX02-0XE0

FMs and CPs for distributed one-way use

Note

You can use all the FMs and CPs released for the ET 200M with the S7-400H distributed and one-way.

FMs and CPs for distributed switched use

Module	Order no.	Release
Communications processor CP341-1 (point-to-point connection)	6ES7341-1AH00-0AE0	hardware version 3 or later
	6ES7341-1AH01-0AE0	hardware version 1 or later with firmware version V1.0.0 or higher
Counter module 350-1	6ES7350-1AH01-0AE0	hardware version 1 or later
Counter module 350-2	6ES7350-2AH00-0AE0	hardware version 2 or later
CAM module FM352	6ES7352-1AH01-0AE0	hardware version 4 or later
Controller module FM 355 C	6ES7355-0VH10-0AE0	hardware version 4 or later
Controller module FM 355 S	6ES7355-1VH10-0AE0	hardware version 3 or later

Glossary

1–out–of–2 system

See **two-channel fault-tolerant system**

Comparison error

An error that may occur while memories are being compared on a fault-tolerant system.

ERROR-SEARCH

An operating mode of the standby CPU of a fault-tolerant system in which the CPU performs a complete self-test.

Fail-safe systems

Fail-safe systems are characterized by the fact that they remain in a safe state when certain failures occur or go directly to another safe state.

Fault-tolerant system

Fault-tolerant system consisting of at least two central processing units (master and standby). The user program is processed in an identical manner in both the master and standby CPUs.

Fault-tolerant systems

Fault-tolerant systems are designed to reduce production stoppages. Availability can be enhanced, for example, by means of component redundancy.

H station

Fault-tolerant station containing two central processing units (master and standby).

I/O, one-sided

We speak of a one-sided I/O when an input/output module can be accessed by only one of the redundant central processing units. It may be single-channel or multi-channel (redundant).

I/O, redundant

We speak of a redundant I/O when there is more than one input/output module available for a process signal. It can be connected as a one-sided or switched I/O. Terminology: “redundant one-sided I/O” or “redundant switched I/O”

I/O, single-channel

We speak of a single-channel I/O when – in contrast to a redundant I/O – there is only one input/output module for a process signal. It can be connected as a one-sided or switched I/O.

I/O, switched

We speak of a switched I/O when an input/output module can be accessed by all of the redundant central processing units on a fault-tolerant system. It may be single-channel or multi-channel (redundant).

Link-up

In the Link-up system mode of a fault-tolerant system the master CPU and the standby CPU compare the memory configuration and the contents of the load memory. If they establish differences in the user program, the master CPU updates the user program of the standby CPU.

Master CPU

The central processing unit that is the first redundant central processing unit to start up. It continues to operate as the master when the redundant link is lost. The user program is processed in an identical manner in both the master and standby CPUs.

Mean time between failures (MTBF)

The average time between two failures and, consequently, a criterion for the reliability of a module or a system.

Mean down time (MDT)

The mean down time essentially consists of the time until error detection and the time required to repair or replace defective modules.

Mean time to repair (MTTR)

Mean time to repair denotes the average repair time of a module or a system – in other words, the time between the occurrence of an error until the error has been rectified.

Redundancy, functional

Redundancy with which the additional technical means are not only constantly in operation but also involved in the scheduled function. Synonym: active redundancy.

Redundant link

A link between the central processing units of a fault-tolerant system for synchronization and the exchange of data.

Redundant mode

In the Redundant system mode of a fault-tolerant system the central processing units are in RUN mode and are synchronized over the redundant link.

Redundant systems

Redundant systems are characterized by the fact that important automation system components are available more than once (redundant). When a redundant component fails, processing of the program is not interrupted.

Self-test

In the case of fault-tolerant CPUs defined self-tests are executed during startup, cyclical processing and when comparison errors occur. These check the contents and the state of the CPU and the I/Os.

Solo mode

In the Solo system mode of a fault-tolerant system the master CPU is in RUN mode, whereas the standby CPU is at STOP, ERROR-SEARCH or DEFECTIVE.

Standby CPU

The redundant central processing unit of a fault-tolerant system that is linked to the master CPU. It goes to STOP when the redundant link is lost. The user program is processed in an identical manner in both the master and standby CPUs.

Stop

With fault-tolerant systems: in the Stop system mode of a fault-tolerant system the central processing units of the fault-tolerant system are in STOP mode.

Synchronization submodule

An interface module to the redundant link on a fault-tolerant system.

Two-channel fault-tolerant system

A fault-tolerant system having two central processing units

Update

In the system mode update a fault-tolerant system the master CPU updates the dynamic data of the standby CPU.

Index

L

M

N

O

P

R

S

T

U

W

Siemens AG
A&D AS E 81

Oestliche Rheinbrueckenstr. 50
D-76181 Karlsruhe
Federal Republic of Germany

From:

Your Name: _

Your Title: _

Company Name: _

Street: _

City, Zip Code _

Country: _

Phone: _

Please check any industry that applies to you:

- ❐ Automotive
- ❐ Chemical
- ❐ Electrical Machinery
- ❐ Food
- ❐ Instrument and Control
- ❐ Nonelectrical Machinery
- ❐ Petrochemical
- ❐ Pharmaceutical
- ❐ Plastic
- ❐ Pulp and Paper
- ❐ Textiles
- ❐ Transportation
- ❐ Other _ _ _ _ _ _ _ _ _ _ _

Remarks Form

Your comments and recommendations will help us to improve the quality and usefulness of our publications. Please take the first available opportunity to fill out this questionnaire and return it to Siemens.

Please give each of the following questions your own personal mark within the range from 1 (very good) to 5 (poor).

1. Do the contents meet your requirements? ☐
2. Is the information you need easy to find? ☐
3. Is the text easy to understand? ☐
4. Does the level of technical detail meet your requirements? ☐
5. Please rate the quality of the graphics/tables: ☐

Additional comments:

_ _
_ _
_ _
_ _
_ _
_ _
_ _
_ _
_ _
_ _
_ _
_ _