SIEMENS

SIMATIC

Automation System S7-400H
Fault-tolerant Systems

Manual

This manual has the order number
6ES7988-8HA11-8BA0

A5E00267695-03

07/2006

Safety Guidelines

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

Danger

indicates that death or severe personal injury will result if proper precautions are not taken.

Warning

indicates that death or severe personal injury may result if proper precautions are not taken.

Caution

with a safety alert symbol indicates that minor personal injury can result if proper precautions are not taken.

Caution

without a safety alert symbol indicates that property damage can result if proper precautions are not taken.

Attention

indicates that an unintended result or situation can occur if the corresponding notice is not taken into account.

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

Qualified Personnel

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

Prescribed Usage

Note the following:

Warning

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

Correct, reliable operation of the product requires proper transport, storage, positioning and assembly as well as careful operation and maintenance.

Trademarks

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

Preface

Purpose of the manual

This manual represents a useful reference and contains information on operating options, functions and technical data of the S7-400H CPU.

For information on installing and wiring those and other modules to install an S7-400H system, refer to the *S7-400 Programmable Controllers, Installation manual.*

Basic knowledge required

A general knowledge of automation technology is considered essential for the understanding of this manual.

We presume that the readership has sufficient knowledge of computers or equipment similar to a PC, such as programming devices, running under the operating system Windows 2000 or XP. An S7-400H is configured using the STEP 7 basic software, and you should thus be familiar in the handling of this software. This knowledge is provided in the *Programming with STEP 7* manual.

In particular when operating an S7-400H system in safety areas, you should always observe the information on the safety of electronic control systems provided in the appendix of the *S7-400 Programmable controllers, Installation* manual.

Validity of the manual

The manual is relevant to the following components:

- CPU 414-4H 6ES7 414-4HJ04-0AB0 withfirmware version V4.0.x or higher
- CPU 417-4H 6ES7 417-4HL04-0AB0, with firmware version V4.0.x or higher

Versions required or order numbers of essential system components

System component	Version required or order number
External master on PROFIBUS DP CP443-5 Extended	Order no. 6GK7 443-5DX03-0XE0, hardware version 1 or higher, and firmware version 5.0 or higher
	Order no. 6GK7 443-5DX04-0XE0, hardware version 1 or higher, and firmware version 6.0.31 or higher
Communication module CP443-1 (Industrial Ethernet, TCP / ISO transport)	6GK7 443-1EX10-0XE0, hardware version 1 or higher, and firmware version V6.7
	6GK7 443-1EX11-0XE0, hardware version 1 or higher, and firmware version V6.7
Communication module CP443-5 Basic (PROFIBUS; S7 communication)	6GK7 443-5FX02-0XE0, hardware version 2 or higher, and firmware version V2.3.2

Notice

There may be further restriction for various modules. Refer to the information in the corresponding product information and FAQs, or in SIMATIC NET News.

Installing the STEP 7 hardware update

In addition to STEP 7, you also need a hardware update. You can download the update files directly from the STEP 7 pages on the Internet. To install the updates, select **STEP 7 -> Configure Hardware** , then select the **Options -> Install Hardware Updates** command.

Certification

For details on certifications and standards, refer to the *S7-400 Programmable Controllers, Module Data* manual, chapter 1.1, Standards and Certifications.

Place of this documentation in the information environment

This manual can be ordered separately under order no. 6ES7988-8HA11-8AA0. It is also supplied in electronic format on your "STEP 7" product CD.

Online Help

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

The Help system can be accessed using various interfaces:

- The **Help** menu contains several commands: **Contents** opens the Help index. The Help on H-systems is found under **Configuring H-Systems**.

- **Using Help** provides detailed instructions on using the Online Help system.

- A context-sensitive Help provides information on the current context, for example, on an open dialog box or an active window. You can call this help by clicking "Help" or using the F1 key.

- The status bar represents a further form of context-sensitive Help. It shows a short description of each menu command when you place the mouse pointer over a command.

- A short info is also shown for the toolbar buttons when you hold the mouse pointer briefly over a button.

If you prefer to read the information of the Online Help in printed form, you can print individual topics, books or the entire Help.

Finding Your Way

To help you find special information quickly, the manual contains the following index tools:

- The manual starts with atable of contents and an index of pictures and tables your manual contains.

- The left column on each page of the chapters provides overview of the contents of each section.

- The appendices are followed by a glossary which defines important special terminology used in this manual.

- At the end of the manual you will find an index which allows quick access to relevant information.

Recycling and Disposal

The S7-400Hsystem contains environmentally compatible materials and can thus be recycled. For environmentally compliant recycling and disposal of your old device, contact a certified recycling company for electronic waste.

Further Support

If you have any technical questions, please get in touch with your Siemens representative or agent responsible.

You will find your contact person at:

`http://www.siemens.com/automation/partner`

You will find a guide to the technical documentation offered for the individual SIMATIC Products and Systems here at:

`http://www.siemens.com/simatic-tech-doku-portal`

The online catalog and order system is found under:

`http://mall.automation.siemens.com`

H/F Competence Center

The H/F Competence Center at our Nuremberg location offers a special workshop with the focus set on redundant SIMATIC S7 automation systems. The H/F Competence Center also offers configuration and commissioning support, and help in finding solutions for problems at your plant.
Phone: +49 (911) 895–4759
Fax: +49 (911) 895–4519
e-mail: HF–CC@siemens.com

Training Centers

Siemens offers a number of training courses to familiarize you with the SIMATIC S7 automation system. Please contact your regional training center or our central training center in D 90327 Nuremberg, Germany for details:

Telephone: +49 (911) 895-3200.

Internet: `http://www.sitrain.com`

Technical Support

You can reach the Technical Suport for all A&D products

- Via the Web formula for the Support Request
  `http://www.siemens.com/automation/support-request`

- Phone: + 49 180 5050 222

- Fax:+ 49 180 5050 223

Additional information about our Technical Support can be found on the Internet pages:
`http://www.siemens.com/automation/service.`

Service & Support on the Internet

In addition to our documentation, we offer our Know-how online on the internet at:

`http://www.siemens.com/automation/service&support`

where you will find the following:

- The newsletter, which constantly provides you with up-to-date information on your products.

- The right documents via our Search function in Service & Support.

- A forum, where users and experts from all over the world exchange their experiences.

- Your local representative for Automation & Drives.

- Information on field service, repairs, spare parts and more under "Services".

Contents

Contents

Figures

Tables

Fault-Tolerant Programmable Logic Controllers

<div style="text-align: right; font-size: large;">**1**</div>

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

In Section	Description	On Page
1.1	Redundant Programmable Logic Controllers in the SIMATIC Series	1-2
1.2	Increasing System Availability	1-4

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 redundant 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 redundant 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.

Fields of application for redundant automation systems

Redundant programmable logic controllers are used in practice with the aim of achieving a higher degree of availability or fault tolerance.
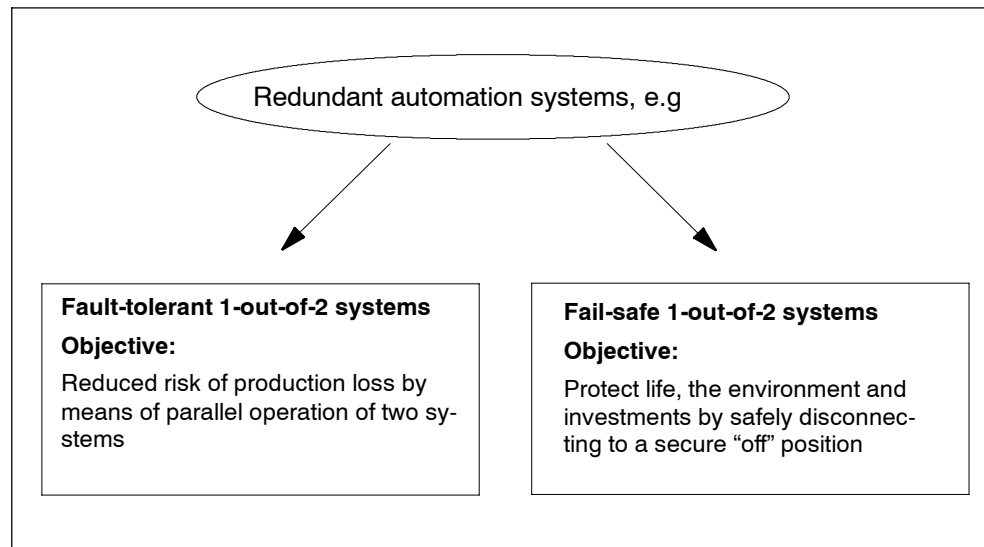
Figure 1-1   Operating objectives of redundant programmable logic controllers

Note the difference between redundant and failsafe systems. An S7-400H represents a redundant automation system which always requires additional measures in order to control safety–relevant processes.

The purpose of redundant automation systems

The objective in using redundant automation systems is to reduce the risk of production losses, regardless whether the losses are caused by an error or as a result of maintenance work.

The higher the costs of down times, the more worthwhile it is to use a redundant system. The generally higher investment costs of redundant systems are quickly returned by he avoidance of production losses.

Software redundancy

In many fields of application, the demands on redundancy quality or the scope of plant units which may require redundant automation systems do not necessarily justify the implementation of a special redundant system. Usually, simple software mechanisms prove sufficient to allow continuation of a failed control process on a substitute system in the event of an error.

The optional "SIMATIC S7 Software Redundancy" software package may be implemented on S7-300 and S7-400 standard systems in order to control processes which tolerate changeover delays to a substitution system in the seconds range, such as water works, water treatment systems or traffic flows.

Redundant I/O

I/O modules are considered redundant when there are two of each and are configured and operated as redundant pairs. The use of redundant I/O returns maximum availability, because such systems will tolerate failure of a CPU and of a signal module, see chapter 8.4.

Redundant I/O are implemented using the blocks of the "functional I/O redundancy" block library.

These blocks are available in the "Redundant IO(V1)" library, under STEP 7\S7_LIBS\RED_IO. For further information on the functionality and use of these blocks, refer to the corresponding online help.

1.2 Increasing System Availability

The S7-400H automation system satisfies the high demands on availability, intelligence and distribution put on state-of-the-art programmable logic controllers. The system provides all functionality required for the acquisition and preparation of process data, including functions for the control, open–loop control and monitoring f aggregates and plants.

Totally integrated systems

The S7-400H automation system and all other SIMATIC components, such as the SIMATIC PCS7 control system, are harmonized. The totally integrated system, ranging from the control room to the sensors and actuators, is a matter of course and guarantees maximum system performance.
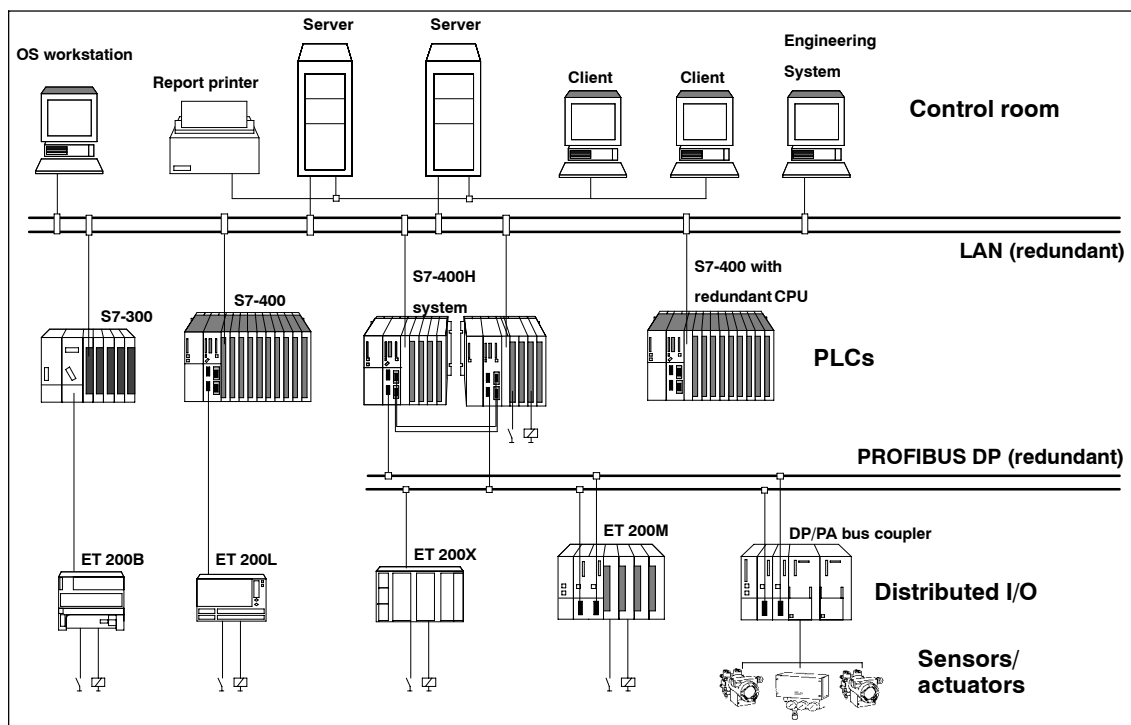
Figure 1-2   Totally integrated automation solutions with SIMATIC

Graduated availability by duplicating components

The redundant structure of the S7-400H ensures availability at all times, i.e., all essential components are duplicated.

This redundant structure includes the CPUs, the power supply modules, and the hardware couplers for both CPUs.

Any further components you may duplicate in order to increase availability are determined by your specific automation process.

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).

Redundant nodes with 1-of-2 redundancy
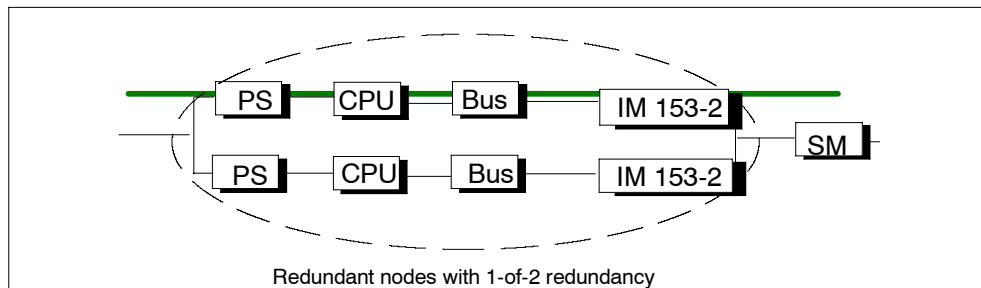
Figure 1-3   Example of redundancy in a network without error

With error

Fig. 1-4 shows that a component may fail without impairing the functionality of the overall system.
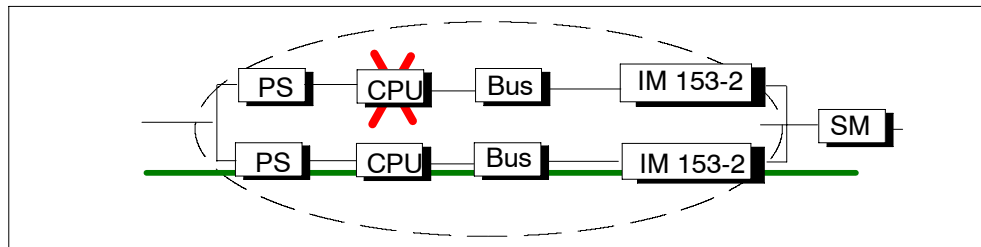
Figure 1-4   Example of redundancy in a 1-of-2 system with error

Failure of a redundant node (total failure)

Fig. 1-5 shows that the system is no longer operable, because both subunits have failed in a 1-of-2 redundant node (total failure).

Redundant nodes with 1-of-2 redundancy
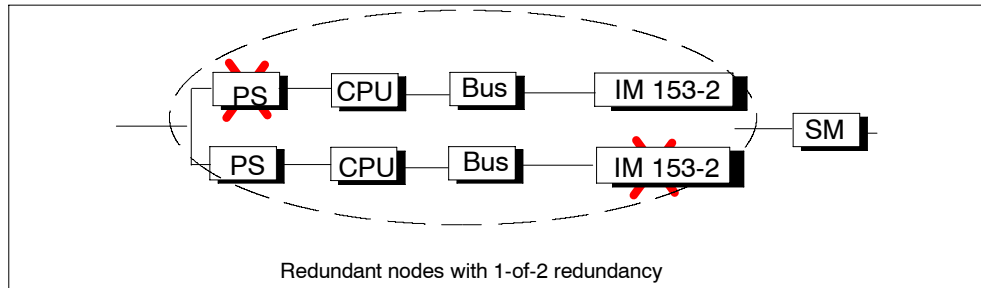
Figure 1-5   Example of redundancy in a 1-of-2 system with total failure

S7-400H Installation Options

<div style="text-align: right;">

2

</div>

The first part of the description deals with the basic configuration of the redundant S7-400H automation system, and with the components of an S7-400H base system. This is continued with the description of the hardware components you can use to expand this base system.

The second part deals with the engineering tools which you are going to use to configure and program the S7-400H. Included is a description of the add–on and extended functions available for the S7-400 base system which you need to create the user program, and to utilize all the properties of your S7-400H in order to increase availability.

In chapter	Description	On Page
2.1	Rules for the assembly of redundant stations	2-3
2.2	Base System of the S7-400H	2-3
2.3	I/O for the S7-400H	2-5
2.4	Communications	2-6
2.5	Configuration and programming tools	2-7
2.6	User program	2-8
2.7	Documentation	2-9

Important information on the configuration

> **Warning**
>
> Open equipment
>
> S7-400 modules are classified as open equipment, i.e. you must install the S7-400 in a cubicle, cabinet or switch room which can only be accessed by means of a key or tool. Such cubicles, cabinets or switch rooms may only be accessed by instructed or authorized personnel.

Fig. 2-1 shows an example of an S7-400H configuration with shared distributed I/O and connection to a redundant system bus. The next pages deal with The HW and SW component required for the installation and operation of the S7-400H.
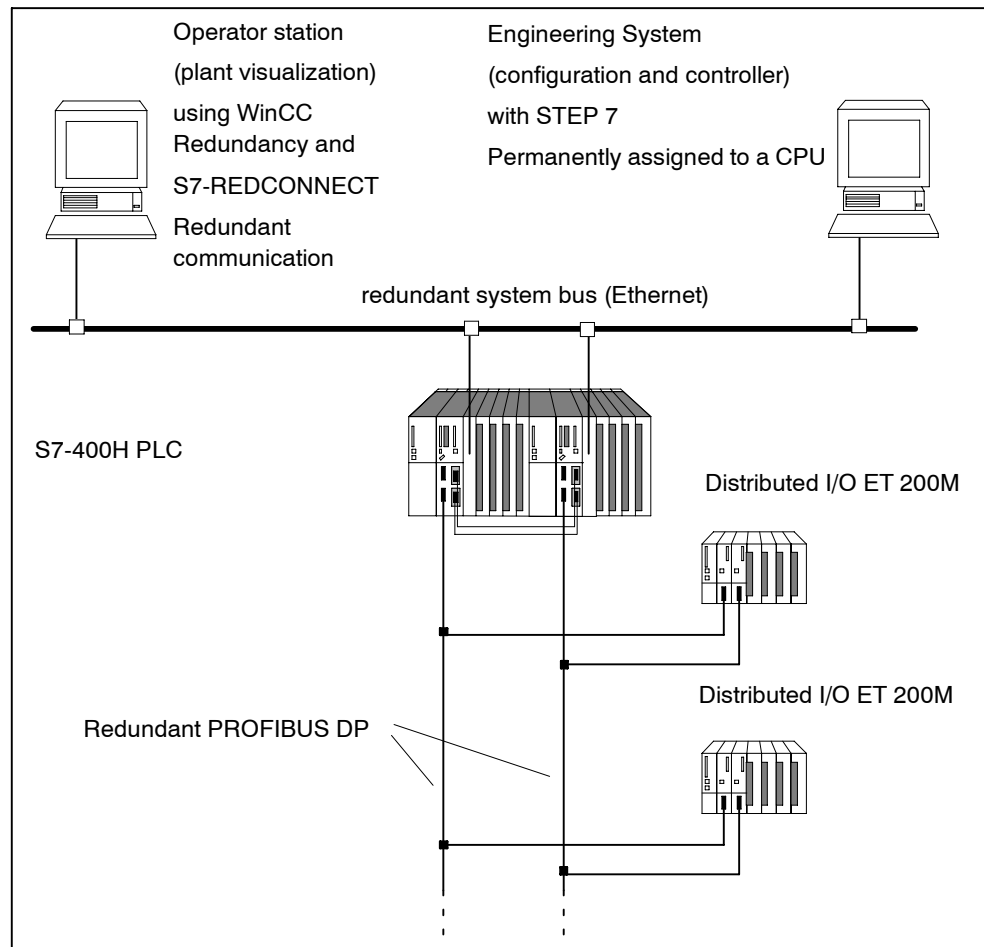
Figure 2-1   Overview

Further information

The components of the S7-400 standard system are also used in the redundant S7-400H programmable logic controller. For detailed information on hardware components for S7-400, refer to the *S7-400 Programmable Controller; Module Data* reference manual.

The rules governing the design of the user program and the use of components laid down for the S7-400 standard system also apply to the redundant S7–400H automation system. Refer to the descriptions in the *Programming with STEP 7* manual, and to the *System Software for S7-300/400; Standard and System Functions* reference manual.

2.1   Rules for the assembly of redundant stations

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

*   The CPUs always have to be inserted in the same slots.

*   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, rather than in expansion racks.

*   Redundantly used modules (for example, CPU 417-4H, DP slave interface module IM 153-2) must be identical, i.e. they must have the same order number, the same version, and the same firmware version.

2.2   Base System of the S7-400H

Hardware of the base system

The base system consists of the hardware components required for a redundant PLCFigure 2-2 shows the components in the installation.

The base system may be expanded with the standard modules of an S7-400. Restrictions only apply the function / communication modules, see the appendix E.
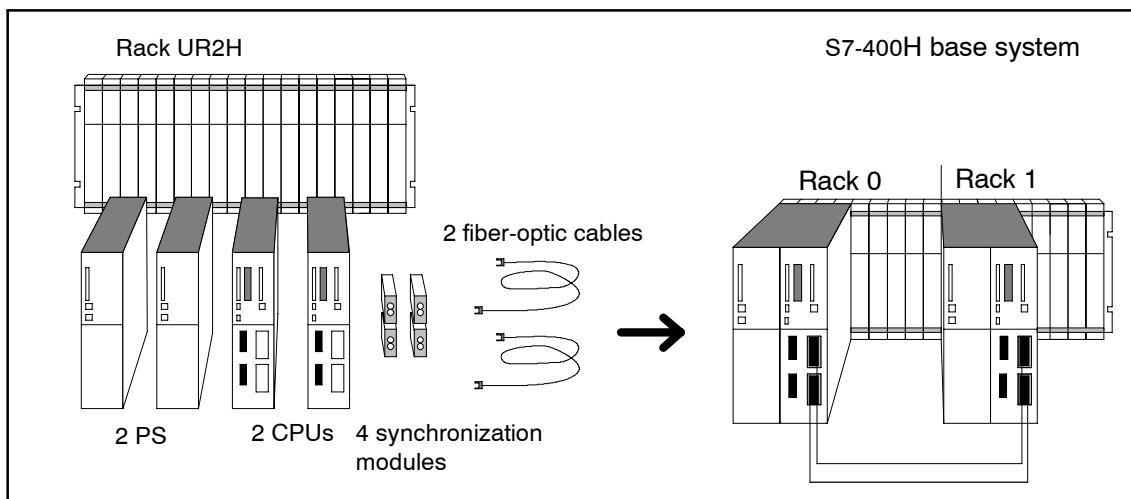
Figure 2-2   Hardware of the S7-400H base system

Central processing units

The two CPUs represent the core components of the S7-400H. Use the switch **on the rear panel** of the CPU to set the rack number. 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.

Rack for S7-400H

The UR2-H rack supports the installation of two separate units with nine slots each, and is suitable for installation in 19" cabinets.

You can also install the S7-400H on two separate racks. The racks UR1 and UR2 are available for this purpose.

Power supply

You require one power supply module from the standard range of the S7-400 for each redundant CPU, or to be more precise, for each of the two units of the S7-400H.

The power supply modules available have rated input voltages of 24 VDC and 120/230 VAC, at an output current of 10 and 20 A.

In order to increase availability of the power supply, you may also use two redundant power supplies in each unit. For this configuration, you should use the PS 407 10 A R power supply module for rated voltages of 120/230 VAC and an output current of 10 A.

Synchronization modules

The synchronization modules which are used to couple the two CPUs are installed in the CPUs and interconnected by means of fiber-optic cables.

There are two types of synchronization modules: one for distances up to 10 m, and one for distances up to 10 km between the CPUs.

The redundant system requires four synchronization modules of the same type. A description of the synchronization modules is found in chapter 13.1.

Fiber-optic cables

The fiber–optic cables are used to interconnect the synchronization modules for the redundant link between the CPUs. They interconnect the two upper, respectively the two lower pairs of the synchronization modules.

The specification of fiber–optic cables which are suitable for use in an S7-400H is found in chapter 13.3.

2.3   I/O Modules for S7-400H

The S7-400H can be equipped with I/O modules of the SIMATIC S7 series.. This I/O can be sued in the following devices:

- central devices
- expansion devices
- as distributed I/O on PROFIBUS DP.

The function modules (FMs) and communication modules (CPs) which are suitable for use in the S7-400H are found in Appendix E.

Versions of the I/O configuration

Versions for the configuration of I/O modules:

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

  With the single-channel, one-sided configuration: single input/output modules. The I/O modules are located in only one unit, and are always addressed by this unit.

  However, the CPUs are interconnected by means of redundancy coupler when operating in redundant mode and thus execute the user program in parallel.

- Single-channel, switched configuration with enhanced availability

  Switched single–channel distributed configurations contain only one set of the I/O modules which can be addressed by both units.

- Redundant dual–channel configuration with maximum availability

  A redundant dual–channel configuration contains two sets of the I/O modules which can be addressed by both units.

Further information

For detailed information on using I/O, refer to chapter 8.

2.4 Communication

The S7-400H supports the following communication methods and mechanisms:

- System bus with Industrial Ethernet

- point-to-point connection

This equally applies to the central and distributed components you can use. Suitable communication modules are listed in appendix E.

Communication availability

You can vary the availability of communications with the S7-400H. The S7-400H supports various solutions to meet your communication requirements. These range from a simple linear network structure to a redundant optical two-fiber loop.

Redundant communication on PROFIBUS or Industrial Ethernet networks is fully supported by the S7 communication functions.

Programming and configuring

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

All communication functions required for redundant communication are integrated in the operating system of the redundant CPU. These functions run automatically in the background, for example, to monitor the communication connection, or to automatically changeover the redundant connection in the event of error.

Further information

For detailed information on communications with the S7-400H, refer to chapter 9.

2.5 Tools for Configuration and Programming

Similar to the S7-400, the S7-400H is also configured and programmed using STEP 7.

You only need to make allowances for slight restrictions when you write the user program However, there are some additional details specific to the redundant configuration. The operating system monitors the redundant components and automatically changes over to the standby components when an error occurs. You have already made the relevant information known to the system in your STEP 7 program.

For detailed information, refer to the Online Help, to chapter 10 and to the appendix D.

Optional Software

All standard tools, engineering tools and Runtime software used in the S7-400 systemare also supported by the S7-400H system.

2.6 The user program

The rules of designing and programming a standard S7-400 system also apply to the S7-400H.

From the viewpoint of user program execution, the S7-400H behaves in exactly the same manner as a standard system. The integral synchronization functions of the operating system are executed automatically in the background. You do not need to configure these functions in your user program.

In redundant operation, the user programs are stored and executed synchronously and event–driven on both CPUs.

However, we offer you various blocks which you can use to tune your program in order to improve its response to any extension of cycle times due to operations such as updates.

Specific Blocks for S7-400H

In addition to the blocks supported the S7-400 and S7-400H systems, the S7-400H software provides further blocks you can use to influence the redundancy functions.

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

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

SFC 90 "H_CTRL" can be used to influence redundant systems as follows:

- You can disable coupling in the master CPU.
- You can inhibit updates in the master CPU.
- You can remove, resume or immediately start a test component of the cyclic self–test.

Notice

Always download these error OBs to the S7-400H CPU: OB 70, OB 72, OB 80, OB 82, OB 83, OB 85, OB 86, OB 87, OB 88, OB 121 and OB 122. If you ignore this, the redundant CPU goes into STOP when an error occurs.

Further information

For detailed information on programming the blocks listed above, refer to the *Programming with STEP 7* manual, and to the *System Software for S7-300/400; System and Standard Functions* reference manual.

2.7   Documentation

2-9

The diagram below provides an overview of the descriptions of the various components and options in the S7-400H Programmable Controllers.

Topic	Documentation
Hardware: Redundancy–capable power supply Rack UR2-H	*S7 standard documentation* Installation Module Specifications Instruction List
IM 153-2	*ET 200M Distributed I/O*
IM 157	*Bus couplers DP/PA-Link and Y-Link*
H-specific programming: H-specific OBs, SFC H-specific expansion of the SSL, events and help on error	*STEP 7 documentation* Programming with STEP 7 V5.4 System and Standard Functions (manual and online Help)
Specifically for redundant systems: Fault-tolerant Systems Configuration Options for S7-400H Getting Started System Modes for S7-400H Link-up and Update I/O, Communications Configuration with the STEP 7 Option Pack Failure and Replacement, System Modification	*S7-400H PLC* Fault-Tolerant Systems (manual and online Help)
Fault-Tolerant Systems Configuring and Programming fail–safe systems Working with S7 F-systems V 5.2	*S7 F/FH Automation Systems* Manual

Figure 2-3   User documentation for redundant systems

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 to a fault.

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

In Section	Description	On Page
3.1	Requirements	3-2
3.2	Configuring Hardware and Starting Up the S7-400H	3-3
3.3	Examples of Fault-Tolerant System Response to Faults	3-5

3.1 Requirements

The following requirements must be met:

Installation of a valid version of the standard STEP 7 software on your PG, see chapter 10.1 .

Modules required for the hardware configuration:

- an S7-400H automation system consisting of:
    - 1 rack, UR2-H
    - 2 power supply modules, PS 407 10A
    - 2 H-CPUs, 414-4H or 417-4H CPUs
    - 4 synchronization modules
    - 2 fiber-optic cables
- an ET 200M distributed I/O device with active backplane bus and
    - 2 IM 153-2
    - 1 digital input module, SM321 DI 16 x DC24V
    - 1 digital output module, SM322 DO 16 x DC24V
- all necessary accessories, such as PROFIBUS cables, etc.

3.2   Hardware installation and S7-400H commissioning

Installing Hardware

To install the S7-400H as shown in Figure 3-1:
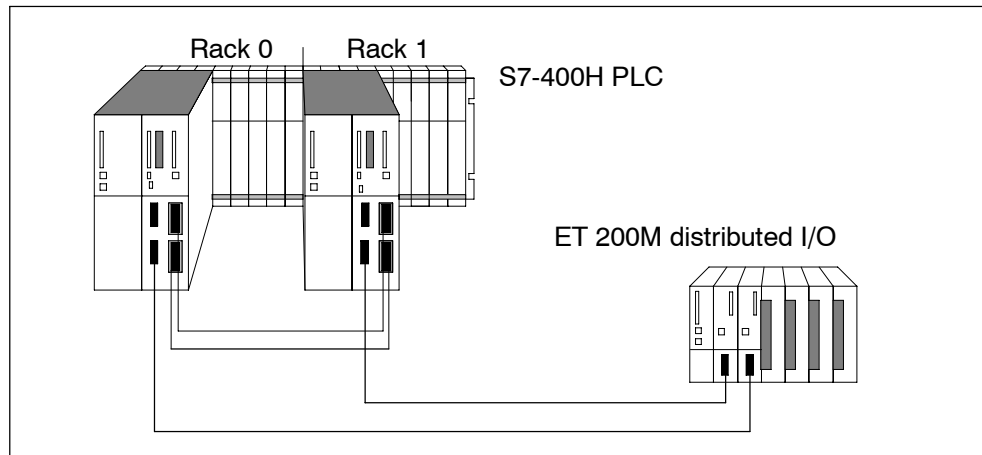
Figure 3-1   Hardware configuration

1. Install both modules of the S7-400H automation system as described in the *S7-400 Programmable Controller, Installation and Module Data* manual.

2. Set the rack numbers using the switch on the rear panel of the CPUs.

    The CPU applies these settings after POWER ON. A faulty rack number setting prevents online access and, under certain circumstances, CPU run also.

3. Install the synchronization modules the CPUs as described in the *S7-400 Programmable Controller, Installation*manual.

4. Connect the fiber–optic cables.

    Always interconnect the upper two, respectively the lower two synchronization modules of the CPUs. Route your fiber–optic cables so that these are safely protected against any damage.

    Always route the fiber–optic cables separately in order to increase availability and protect them from any double error which may be caused by failure of both fiber–optic circuits.

    Always connect the fiber–optic cables to the CPUs before you switch on the power supply or the system, because otherwise both CPUs may process the user program in master mode.

5. Configure the distributed I/O as described in the *ET 200M Distributed I/O Device* manual.

6. Connect the PG to the first redundant CPU, namely CPU0. This CPU will be the master of your S7-400H.

7. The high-quality RAM test which is performed after power on takes approx. 10 minutes. The CPU can not be accessed and the STOP LED flashes for the duration of this test. A further test after the next POWER ON will be discarded if the CPU is equipped with a backup battery.

Commissioning the S7-400H

To commission the S7-400H

1. In SIMATIC Manager, open the sample project"HProjekt". The configuration corresponds with the HW configuration described in "Requirements".

2. To open the hardware configuration of the project, right–click the "Hardware" object, and then select **Object ▸ Open** from the shortcut menu. If your configuration matches, continue with step 6.

3. If your hardware configuration does not match the project, for example, with respect to module types, MPI addresses or DP address, edit and save the project accordingly. For further information, refer to the basic help of SIMATIC Manager.

4. Open the user program in the "S7 program" folder.

   In the offline view, this folder is always assigned to CPU0. The user program is executable with the described hardware configuration, and controls the LED bar graph on the digital output module accordingly.

5. If necessary for your hardware configuration, edit the user program and the save it, for example.

6. Select **PLC ▸ Download** to download the user program to CPU0.

7. Start up the S7-400H automation system by setting the mode selector switch of CPU0 to RUN. The set the selector switch at CPU1.to RUN. The CPU performs a restart and calls OB100.

   Result: CPU0 starts up as the master CPU and CPU1 as the standby CPU. After the standby CPU is coupled and updated, your S7-400H assumes the redundant state and executes the user program and controls the LED bar graph on the digital output module accordingly.

Note

You may also start up and stop the S7-400H automation system usingSTEP 7. For further information, refer to the Online Help.

A cold start is always initiated using the PG command "Cold start". To do so, the CPU must be in STOP, and the mode selector switch must be set to RUN. OB102 will be called in the cold start routine.

3.3 Examples of the reaction of the redundant system to faults

Example 1: Failure of a CPU or of a power supply

Initial situation: The S7-400H is in redundant mode.

1. Simulate a CPU0 failure by turning off the power supply.

   Result: The LEDs REDF, IFM1F and IFM2F light up on CPU1. CPU1 goes into stand–alone mode and continues to process the user program.

2. Turn the power supply back on.

   Result:

   – CPU0 performs an automatic LINK-UP and UPDATE.

   – CPU0 changes to RUN, and now operates in standby mode.

   – The S7-400H now operates in redundant mode.

Example 2: Failure of a fiber–optic interface

Initial situation: The S7-400H is in redundant mode. The mode selector switch of the CPUs are set to RUN..

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 up at both CPUs. The standby CPU goes into STOP. The master CPU continues operation in stand–alone mode.

2. Reconnect the fiber-optic cable.

3. Restart the original standby CPU (CPU1), which is now at STOP, by means of STEP 7 "operating status", for example.

   Result:

   – CPU1 performs an automatic LINK-UP and UPDATE.

   – The S7-400H resumes redundant mode.

Installation of a CPU 41x-H

4

Chapter Overview

4.1 Control and display elements of the CPUs

Control and display elements of the CPU 414-4H/417-4H
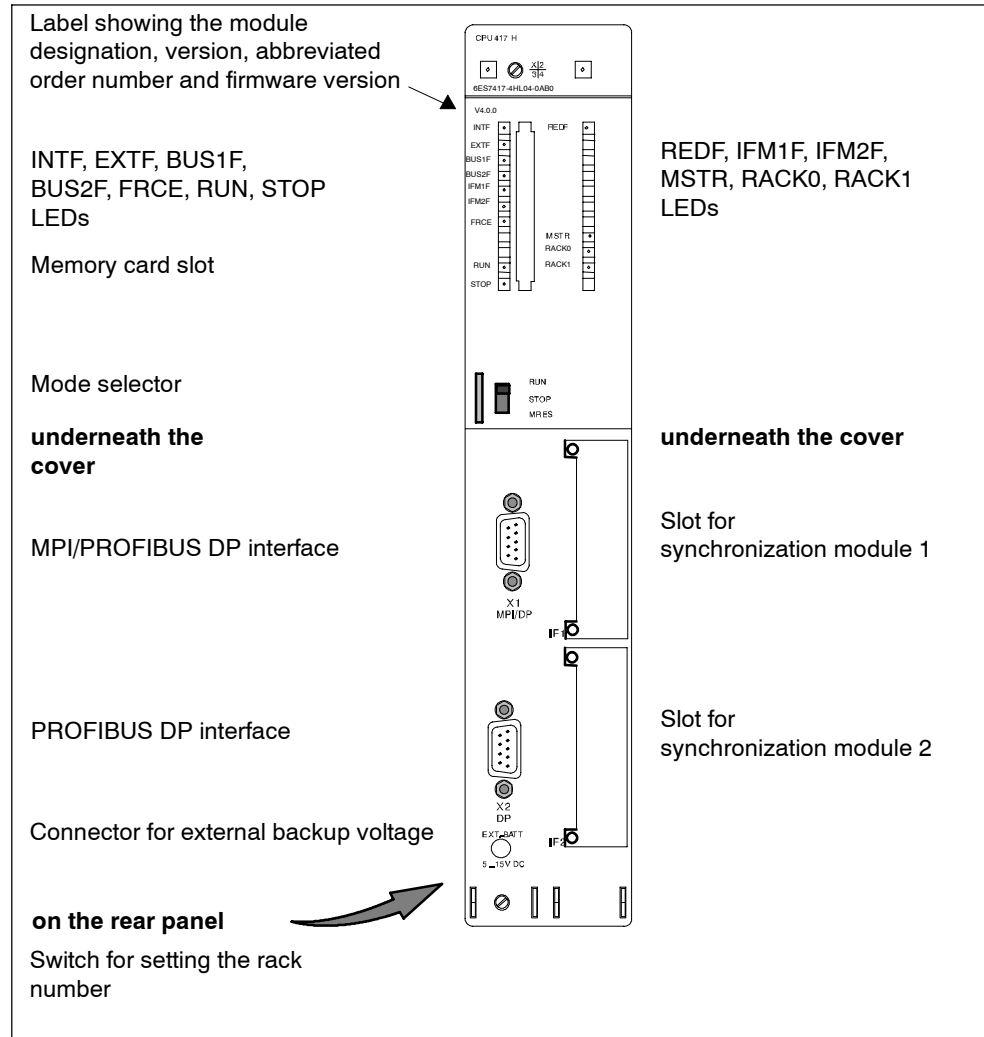
Figure 4-1   Layout of the control and display elements of
CPU 414-4H/417-4H

LED displays

Table 4-1 shows an overview of the LEDs on the various CPUs.

Chapter 4.2 and 4.3 deals with the errors and states indicated by these LEDs.

Table 4-1   LED displays of the CPUs

LED	Color	Meaning
INTF	red	Internal fault
EXTF	red	External fault
FRCE	yellow	Active force request
RUN	green	RUN mode
STOP	yellow	STOP mode
BUS1F	red	Bus error at MPI/PROFIBUS DP interface 1
BUS2F	red	Bus error at PROFIBUS DP interface 2
MSTR	yellow	CPU controls the process
REDF	red	Loss of redundancy/redundancy error
RACK0	yellow	CPU in rack 0
RACK1	yellow	CPU in rack 1
IFM1F	red	Error at synchronisation module 1
IFM2F	red	Error at synchronisation module 2

Mode selector switch

You can use the mode selector switch to set the current operating mode of the CPU. The mode selector switch is a rocker switch with three positions.

Chapter 4.5 deals with the functions of the mode selector switch.

Slot for Memory Cards

You can insert a memory card in this slot.

There are two types of memory card:

- RAM cards

    You can expand the load memory of a CPU with the RAM card.

- FLASH cards

    A FLASH card can be used for fail–safe backup of the user program and data without backup battery. You can program the FLASH card either on the PG or in the CPU. The FLASH card also expands CPU load memory.

    For detailed information, refer to on memory cards, refer to chapter 4.8.

Slot for interface modules

You can insert an H-sync module into this slot.

MPI/DP Interface

Devices you can connect to the MPI of the CPU, for example:

- Programming devices

- Operation and monitoring devices

- Further S7-400 or S7-300 PLCs, see chapter 4.9.

Use bus connectors with angled cable exit, see the *S7–400 Programmable Controller, Installation*, chapter 7

The MPI interface can be configured for operation as DP master and thus as PROFIBUS DP interface for up to 32 DP slaves.

PROFIBUS-DP Interface

The PROFIBUS DP interface supports the connection of distributed I/O, PGs and OPs.

Setting the rack number

Use the selector switch on the rear panel of the CPU to set the rack number. The switch has two positions, namely 1 (up) and 0 (down). One CPU is allocated rack number 0, and the partner CPU is assigned rack number 1. Default setting of both CPUs is rack number 0.

Connecting an external backup voltage to the ”EXT. BATT.” socket

The S7-400H power supply modules support the use of two backup batteries. This allows you to:

- backup the user program stored in a RAM module,
- maintain retentivity of flags, timers, counters, system data and data in dynamic data blocks, and
- buffer the internal clock.

You can achieve the same effect by connecting an auxiliary voltage between 5 VDC and 15 VDC to the ”EXT. BATT.” socket of the CPU.

Properties of the ”EXT. BATT.” input:

- Protection against polarity reversal
- Short-circuit current limiting to 20 mA

An auxiliary voltage is connected to the ”EXT. BATT” input y means of a cable with a 2.5 mm jack as shown in the figure below. Observe the polarity of the jack.
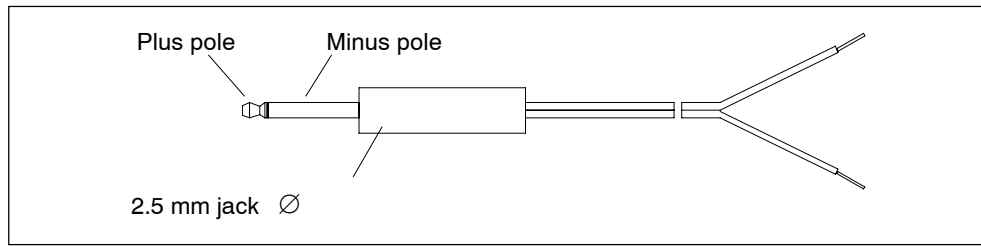
Note

When you replace a power supply module and want to backup the user program and data stored in RAM while doing so, you should connect an auxiliary power supply to the ”EXT. BATT.” input as mentioned earlier.

4.2 Monitoring functions of the CPU

Monitoring functions and error messages

The hardware and operating system of the CPU provide monitoring functions to ensure proper operation and defined reactions to errors. Various errors may also trigger a reaction in the user program.

The table below provides an overview of possible errors and their causes, and the corresponding reactions of the CPU.

Each CPU also provides various test and information functions which you can call in STEP 7.

Error class	Cause of error	Reaction of the operating system	Error LED
Access error	Module failure (SM, FM, CP)	LED "EXTF" remains lit until the error is eliminated. In SMs: • Call of OB122 • Entry in the diagnostics buffer • In the case of input modules: Entry of "null" for the date in the accumulator or the process image In the case of other modules: • Call of OB122	EXTF
Timeout error	• The runtime of the user program (OB1 and all interrupts and error OBs) exceeds the specified maximum cycle time. • OB request error • Overflow of the start information buffer • Time-of-day error interrupt	LED "INTF" remains lit until the error is eliminated. Call of OB80l If the OB is not loaded: The CPU goes into STOP.	INTF
Power supply module(s) error (not power failure)	In the central or expansion rack: • At least one backup battery in the power supply module is low. • The backup voltage is missing. • The 24 V supply to the power supply module has failed.	Call of OB81 If the OB is not loaded: The CPU continues to run.	EXTF
Diagnostics interrupt	An I/O module which supports interrupts reports a diagnostics interrupt.	Call of OB82 If the OB is not loaded: The CPU goes into STOP.	EXTF
Removal/insertion interrupt	Removal or insertion of an SM, and insertion of a wrong module type.	Call of OB83 If the OB is not loaded: The CPU goes into STOP.	EXTF
CPU hardware error	• A memory error was detected and eliminated • Redundancy coupling: data transfer errors.	Call of OB84 If the OB is not loaded: The CPU remains in RUN.	INTF
Program execution error	• Priority class is called, but the corresponding OB is not available. • In the case of an SFB call: missing or faulty instance DB. • Process image update error	Call of OB85 If the OB is not loaded: The CPU goes into STOP.	INTF
			EXTF

Error class	Cause of error	Reaction of the operating system	Error LED
Failure of a rack/station	• Power failure in an expansion rack • Failure of a DP segment • Failure of a coupling segment: missing or defective IM, interrupted cable	Call of OB86 If the OB is not loaded: The CPU goes into STOP.	EXTF
Execution cancelled	Execution of a program block was canceled. Possible reasons for the cancellation are: • Nesting depth of parenthesis above maximum • Nesting depth of Master Control Relay above maximum • Nesting depth of synchronization errors above maximum • Nesting depth of block calls (U stack) above maximum • Nesting depth of block calls (B stack) above maximum • Error allocating local data	Call of OB88 If the OB is not loaded: The CPU goes into STOP.	INTF
Programming error	User program error: • BCD conversion error • Range length error • Range error • Alignment error • Write error • Timer number error • Counter number error • Block number error • Block not loaded	Call of OB121 If the OB is not loaded: The CPU goes into STOP.	INTF
MC7 code error	Error in the compiled user program, for example, illegal OP code or jump over the end of the block	The CPU goes into STOP. Restart or CPU memory reset required.	INTF

4.3   Status and error displays

LEDs RUN and STOP

The RUN and STOP LEDs provide information about the active CPU operating status.

LED		Meaning
RUN	**STOP**	
H	D	CPU is in RUN.
D	H	CPU is in STOP. The user program is not executed. Cold restart/ warm restart is possible. If the STOP status was triggered by an error, the error indicator (INTF or EXTF) is also set.
B 2 Hz	B 2 Hz	CPU status = DEFECT. The INTF, EXTF and FRCE LEDs also flash.
B 0.5 Hz	H	STOP status has been triggered by a test function.
B 2 Hz	H	A cold restart / warm restart was initiated. The restart / warm restart may take a minute or longer, depending on the length of the called OB. If the CPU still does not go into RUN, there might be an error in the system configuration.
D	B 2 Hz	Self–test with unbuffered POWER ON is busy. The self–test may take up to 10 minutes. CPU memory reset is busy
x	B 0.5 Hz	The CPU requests a memory reset.
B 0.5 Hz	B 0.5 Hz	Debugging mode

D = LED is dark; H = LED lights up; B = LED flashes with the specified frequency; x = LED status is irrelevant

LEDs MSTR, RACK0 and RACK1

The three LEDs, MSTR, RACK0 and RACK1 provide information about the mounting rack number configured at the CPU, and show which CPU controls the switched I/O modules.

LED			Meaning
MSTR	**RACK0**	**RACK1**	
H	x	x	CPU controls the switched I/O
x	H	D	CPU on rack number 0
x	D	H	CPU on rack number 1

D = LED is dark; H = LED is lit; x = LED status is irrelevant

LEDs INTF, EXTF and FRCE

The three LEDs, INTF, EXTF and FRCE, provide information about errors and special events in user program execution.

LED			Meaning
INTF	**EXTF**	**FRCE**	
H	x	x	An internal error was detected (programming or parameter assignment error).
x	H	x	An external error was detected, i.e. an error not triggered by the CPU module.
x	x	H	A force request is busy.

H = LED is lit; x = LED status is irrelevant

LEDs BUSF1 and BUSF2

The LEDs BUSF1 and BUSF2 indicate errors at the MPI/DP and PROFIBUS DP interfaces.

LED		Meaning
BUS1F	**BUS2F**	
H	x	An error was found at the MPI/DP interface.
x	H	An error was found at the PROFIBUS DP interface.
B	x	DP master: one or more slaves at PROFIBUS DP interface 1 are not responding. DP slave: not addressed by the DP master
x	B	DP master: one or more slaves at PROFIBUS DP interface 2 are not responding. DP slave: not addressed by the DP master

H = LED is lit; B = LED flashes; x = LED status is irrelevant

LEDs IFM1F and IFM2F

The LEDs IFM1F and IFM2F indicate errors of the first and second module interfaces.

LED		Meaning
IFM1F	**IFM2F**	
H	x	An error was found at module interface 1.
x	H	An error was found at module interface 2.

H = LED is lit; x = LED status is irrelevant

LED REDF

The LED REDF indicates specific system states and redundancy errors.

REDF LED	System status	Marginal conditions
B 0.5 Hz	Coupling	–
B 2 Hz	Updating	–
D	Redundant (CPUs are redundant)	No redundancy error
H	Redundant (CPUs are redundant)	There is an I/O redundancy error: • Failure of a DP master, or partial or total failure of a DP master system • Loss of redundancy on the DP slave
	All system states, except redundancy, coupling, updating	–

D = LED is dark; H = LED is lit; B = LED flashes at the set frequency

Diagnostics Buffer

You can troubleshoot the error by reading out its precise cause from the diagnostics buffer using STEP 7 (PLC –> Module Information).

4.4 Reading service data

Requirements

Installation of STEP 7 version 5.3 or higher.

Use case

If you need to contact our Customer Support due to a service event, the department may require specific diagnostics information on the CPU status of your system. This information is stored in the diagnostics buffer and in the actual service data.

Select the "PLC –> Save service data" command to read this information and save the data to two files You can then send these to Customer Support.

Please note:

• If possible, save the service data immediately after the CPU goes into STOP or the synchronization of a redundant system was lost.

• Where a redundant system is concerned, always save the service data of both CPUs, i.e. including those of the CPU which is still in RUN after the loss of a synchronization.

Procedure

1. Select the PLC –> Save service data" command

   In the next dialog box, select the file path and the file names.

2. Save the files.

3. Forward these files to Customer Support on request.

4.5 Mode selector switch

Function of the mode selector switch

This switch can be used to set the CPU into RUN and STOP, or the reset CPU memory. STEP 7 offers further options of changing the mode.

Positions

The mode selector switch is a rocker switch. Figure 4-2 shows the positions of the switch.
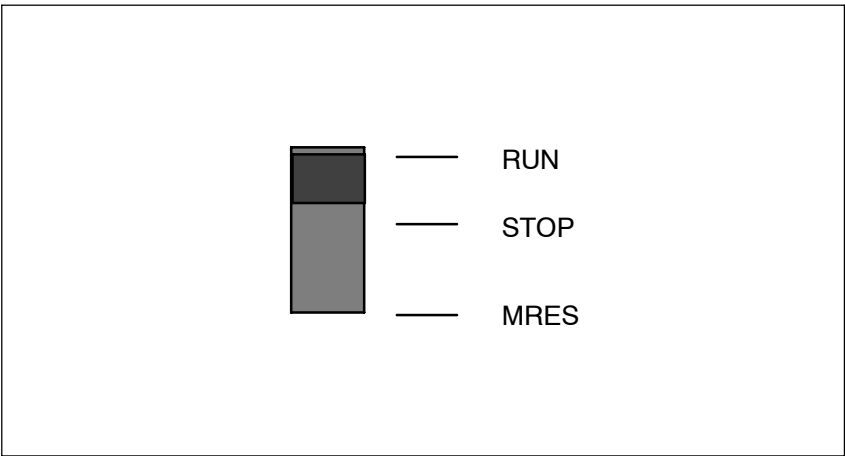
Figure 4-2   Positions of the mode selector switch

Table 4-2 provides details on the switch positions. If an error or a startup error has occurred, the CPU will either go into STOP or maintain this mode, irrespective of the position of the mode selector switch.

Table 4-2   Positions of the mode selector switch

Position	Comments
RUN	If there are no errors preventing startup and no other errors have occurred, the CPU will go into RUN and then execute the user program or remain idle. It is thus possible to access the I/O.
STOP	The CPU does not execute the user program, and the output modules are disable by default.
MRES (CPU memory reset; master reset)	Rocker switch position with pushbutton action, used to reset CPU memory, see chapter 4.7.

4.6 Protection Levels

You can define a security level for your project in order to prevent unauthorized access to the CPU programs. The objective of these security settings is to grant a user access to specific PG functions which are not protected by password, and to allow this user to execute these functions on the CPU. When logged on with password, the user may execute all PG functions.

Setting security levels

You can set the CPU security levels 1 to 3 in STEP 7 under Configure Hardware.

If you do not know the password, you can clear the security setting by means of a manual CPU memory reset using the mode selector switch. The CPU may not contain a Flash card when you perform such an operation.

Table 4-3 lists the security levels of an S7-400 CPU.

Table 4-3   CPU security levels

CPU function	Security level 1	Security level 2	Security level 3
Visualization of the block list	Access granted	Access granted	Access granted
Monitoring tags	Access granted	Access granted	Access granted
Module status STACKS	Access granted	Access granted	Access granted
Control and monitoring functions	Access granted	Access granted	Access granted
S7 communication	Access granted	Access granted	Access granted
Reading the time	Access granted	Access granted	Access granted
Setting the time	Access granted	Access granted	Access granted
Status block	Access granted	Access granted	Password required
Upload to PG	Access granted	Access granted	Password required
Download to CPU	Access granted	Password required	Password required
Deleting blocks	Access granted	Password required	Password required
Compressing memory	Access granted	Password required	Password required
Download of the user program to a Memory Card	Access granted	Password required	Password required
Controlling selection	Access granted	Password required	Password required
Controlling tags	Access granted	Password required	Password required
Breakpoint	Access granted	Password required	Password required
Clear breakpoint	Access granted	Password required	Password required

Table 4-3   CPU security levels, continued

CPU function	Security level 1	Security level 2	Security level 3
Memory Reset	Access granted	Password required	Password required
Force	Access granted	Password required	Password required

4.7   Operating Sequence for Memory Reset

Use case A: You want to download a new user program to the CPU.

1. Set the switch to STOP position.

    Result: The STOP LED is lit.

2. Toggle the switch to MRES, and hold it in this position. This selector switch position has a pushbutton action contact.

    Result: The STOP LED is dark for a second, light for a second, dark for a second and then remains on.

3. Release the switch, return it to MRES within the next three seconds, and then release it again.

    Result: The STOP LED flashes for a duration of at least 3 seconds at 2 Hz (CPU memory reset is being executed), and then lights up continuously

Use case B: The STOP LED flashing slowly at 0.5 Hz indicates that the CPU is requesting a memory reset (system memory reset request, after a memory card has been removed or inserted, for example).

Toggle the switch to MRES, and then release it again.

Result: The STOP LED flashes for a duration of at least 3 seconds at 2 Hz while the CPU memory reset is being executed, and then the LED lights up continuously.

Sequence of a CPU memory reset

Sequence of the CPU for a memory reset:

*   The CPU deletes the user program from RAM.

*   The CPU deletes the user program from load memory. This process deletes the program from the on–board RAM and from any RAM Card. The user program elements stored on Flash card will not be deleted if you have expanded load memory with such a card.

*   The CPU resets all counters. flags and timers, but not the time–of–day.

*   The CPU tests its hardware.

*   The CPU sets its parameters to default values.

*   When a FLASH Card is inserted, the CPU continues after its memory reset by copying the user program and the system parameters from the Flash card to RAM.

Data retained after a CPU memory reset...

The following values are retained after a CPU memory reset:

- The content of the diagnostics buffer

  If you had not inserted a flash card during CPU memory reset, the CPU resets the capacity of the diagnostics buffers to its default setting of 120 entries, i.e. the most recent 120 entries will be retained in the diagnostics buffer.

  You can read the content of the diagnostics buffer using STEP 7.

- The MPI interface parameters. These define the MPI address and the highest MPI address. Note the special features shown in the table below.

- The time

- The status and value of the operating hours counter

Special feature: MPI parameters

The MPI parameters take an exceptional position during CPU memory reset. The table below lists the MPI parameter which are valid after a CPU memory reset.

CPU memory Reset	MPI parameters...
with inserted FLASH Card	...stored on the FLASH Card are valid
without inserted FLASH Card	...in the CPU are retained and thus valid

Cold restart

- A cold restart initializes the process image, all flags, timers, counters and data blocks with the start values stored in load memory, regardless whether these data were configured as being retentive or not.

- Program execution resumes with OB1, or with OB102.

Restart (warm restart)

- A warn restart resets the process image and the non–retentive flags, timers and counters.

  Retentive flags, timers, counters and all data blocks retain their last valid value.

- Program execution resumes with OB1, or with OB101.

- A warm restart after power failure is only possible if memory is backed up.

Operating sequence for restart t/warm restart

1. Set the switch to STOP position.

   Result: The STOP LED lights up.

2. Set the switch to RUN position.

   Result: The STOP LED is dark, the RUN LED is lit.

Whether the CPU performs a restart or a hot restart is determined by its configuration.

Operating sequence for cold restart

A cold start is always initiated using the PG command "Cold start". To do so, the CPU must be in STOP, and the mode selector switch must be set to RUN.

4.8 Expanding Load Memory with Memory Cards

Order Numbers

The order numbers for memory cards are listed in the technical data at the end of this chapter.

Design of a Memory Card

The size of a Memory Card corresponds with that of a PCMCIA card. It is inserted into a front–panel slot of the CPU.
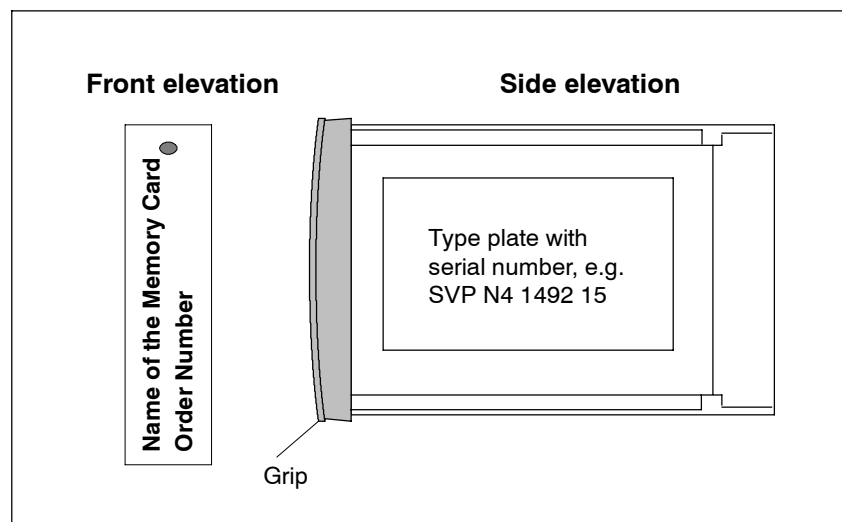
Figure 4-3   Design of the memory card

Function of the Memory Card

The memory card and an integrated memory section on the CPU together form the loading memory of the CPU. During operation, the loading memory contains the complete user program, including the comments, the symbols and special additional information that enables back-translation of the user program as well as all module parameters.

Data stored on Memory Card

The following data can be stored on Memory Card:

- The user program, i.e. the OBs, FBs, FCs, DBs and system data

- Parameters that determine the behavior of the CPU

- Parameters that determine the behavior of the I/O modules.

- As of STEP 7 V5.1, all project files on suitable Memory Cards.

Types of Memory Cards for the S7-400

Two types of memory card are used for the S7-400:

- RAM cards

- Flash cards (FEPROM cards)

Note

Memory cards unknown to the system can not be used in the S7-400.

Which type of Memory Card to use

Whether you use a RAM card or a Flash card depends on your application.

Table 4-4   Types of memory cards

If you ...	...Then
also want to be able to edit your program in RUN,	use a **RAM card**
want to keep a permanent backup of your user program on Memory Card when power is off, i.e. without backup battery or outside the CPU,	use a **Flash card**

RAM Card

Insert the RAM Card for the download of the user program to the CPU. Download the user program in STEP 7 by selecting "PLC –> Download user program to Memory Card".

You can download the entire user program or the separate elements such as FBs, FCs, OBs, DBs, or SDBs to load memory in STOP or in RUN.

When you remove the RAM card from the CPU, the information stored on it will be lost. The RAM card is not equipped with an integrated backup battery.

If the power supply is equipped with an operational backup battery, or the CPU is supplied with an external backup voltage at the "EXT. BATT." input, the RAM Card contents are retained when power is switched off, provided the RAM card remains inserted in the CPU and the CPU remains inserted in the rack.

FLASH Card

If you use a Flash card, there are two ways of loading the user program:

- Use the mode selector switch to set the CPU to STOP. Insert the FLASH Card into the CPU, and then download the suer program in STEP 7 to the Flash card by selecting "PLC –> Download user program to Memory Card".

- Load the user program into the Flash card in offline mode at the programming device/programming adapter, and then insert the Flash card into the CPU.

The FLASH card is a non–volatile memory, i.e. its data are retained when the it is removed from the CPU or your S7-400 is being operated without backup voltage (without a backup battery in the power supply module or external backup voltage at the "EXT. BATT." input of the CPU).

You always download the full user program to a FLASH Card.

Downloading additional user program elements

You can download further elements of the user program from the PG to the integrated load memory of the CPU. Note that the content of this memory area will be deleted if the CPU performs a memory reset, i.e. load memory is updated with the user program stored on Memory Card after a CPU memory reset.

Memory Card capacity requirements

The capacity of your memory card is determined by the scope of the user program.

Determining Memory Requirements using SIMATIC Manager

You can view the block lengths offline by selecting the "Properties - Block folder offline" dialog box (Blocks –> Object Properties –> Blocks tab).

The offline view shows the following lengths:

*   Size (sum of all blocks, without system data) in load memory of the PLC

*   Size (sum of all blocks, without system data) in RAM of the PLC

Block lengths from the engineering device (PG/PC) are not shown in the properties of the block container.

Block lengths are shown in "byte" units.

The following values are shown in the block properties:

*   Required local data volume: length of local data in bytes

*   MC7: length of the MC7 code in bytes

*   Length of DB user data

*   Length in load memory of the PLC

*   Lengths in RAM of the PLC (only if hardware assignment is known.)

The views always show these block data, regardless whether it is located in the window of an online view or of an offline view..

When a block container is opened and "View Details" is set, the project view always indicates RAM requirements, regardless whether the block container appears in the window of an online or offline view.

You can add up the block lengths by selecting all relevant blocks. SIMATIC Manager outputs the total length of the selected blocks in its status bar.

The view does not indicate the lengths of blocks (VATs, for example) which can not be downloaded to the PLC.

Block lengths on the engineering system (PG/PC) are not shown in the Details view.

Technical data

Name	Order number	Current consumption at 5 V	Backupcurrents
MC 952 / 256 KB / RAM	6ES7 952-1AH00-0AA0	typ. 35 mA max. 80 mA	typ. 1 μA max. 40 μA
MC 952 / 1 MB / RAM	6ES7 952-1AK00-0AA0	typ. 40 mA Max. 90 mA	typ. 3 μA max. 50 μA
MC 952 / 2 MB / RAM	6ES7 952-1AL00-0AA0	typ. 45 mA max. 100 mA	typ. 5 μA max. 60 μA
MC 952 / 4 MB / RAM	6ES7 952-1AM00-0AA0	typ. 45 mA max. 100 mA	typ. 5 μA max. 60 μA
MC 952 / 8 MB / RAM	6ES7 952-1AP00-0AA0	typ. 45 mA max. 100 mA	typ. 5 μA max. 60 μA
MC 952 / 16 MB / RAM	6ES7 952-1AS00-0AA0	typ. 45 mA max. 100 mA	typ. 5 μA max. 60 μA
MC 952 / 1 MB / 5-V Flash	6ES7 952-1KK00-0AA0	typ. 40 mA Max. 90 mA	–
MC 952 / 2 MB / 5-V Flash	6ES7 952-1KL00-0AA0	typ. 50 mA max. 100 mA	–
MC 952 / 4 MB / 5-V Flash	6ES7 952-1KM00-0AA0	typ. 40 mA Max. 90 mA	–
MC 952 / 8 MB / 5-V Flash	6ES7 952-1KP00-0AA0	typ. 50 mA max. 100 mA	–
MC 952 / 16 MB / 5V Flash	6ES7 952-1KS00-0AA0	typ. 55 mA max. 110 mA	–
MC 952 / 32 MB / 5-V Flash	6ES7 952-1KT00-0AA0	typ. 55 mA max. 110 mA	–
MC 952 / 64 MB / 5-V Flash	6ES7 952-1KY00-0AA0	typ. 55 mA max. 110 mA	–
Dimensions W $\times$ H $\times$ D (in mm)		7.5 $\times$ 57 $\times$ 87	
Weight		max. 35 g	
EMC protection		Ensured by design measures	

4.9   Multipoint Interface (MPI)

2Connectable Devices

You can, for example, connect the following nodes to the MPI:

- Programming devices (PG/PC)
- Operating and monitoring devices (OPs and TDs)
- Additional SIMATIC S7 PLCs

Some connectable devices take the 24 V supply from the interface. This voltage is non–isolated.

PG/OP-CPU Communication

A CPU is capable of handling several online connections to PGs/OPs in parallel. By default, one of these connections is always reserved for a PG, and one for an OP/operation and monitoring device.

CPU-CPU Communication

CPUs exchange data by means of S7 communication.

For further information, refer to the Programming with STEP *7*” manual.

Connectors

Always use bus connectors with an angular cable exit for PROFIBUS DP or PG cables to connect devices to the MPI (see *Installation Manual*, Chapter 7).

Using the MPI interface as DP interface

You can also configure the MPI interface for operation as DP interface. To do so, reconfigure the MPI interface under STEP 7 in SIMATIC Manager. This feature can be used to configure a DP segment with up to 32 slaves.

4.10   PROFIBUS DP Interface

Connectable Devices

You may connect any slave which is compliant with DP standard to the PROFIBUS DP interface.

Here, the CPU is the DP master that is connected via the field bus PROFIBUS DP with the passive slave stations or, in the standalone mode, other DP masters.

Various compatible devices take the 24 V supply from the interface. This voltage is non–isolated.

Connectors

Always use bus connectors for PROFIBUS DP and PROFIBUS cables to connect devices to the PROFIBUS DP interface (*see he Installation manual,* chapter 5).

Redundant mode

In redundant mode, the PROFIBUS DP interfaces have the same parameters.

4.11   Overview of the parameters of the S7-400 CPUs

Default values

You can determine the CPU-specific default values by selecting "Configuring Hardware" in STEP 7.

Parameter blocks

The reactions and properties of the CPU are set at the parameters which are stored in system data blocks. The CPUs have a defined default setting. You can modify these default values by editing the parameter in HW Config.

The list below provides an overview of the configurable system properties of the CPUs.

- General properties, such as the CPU name

- Startup

- Cycle/clock memory, for example, the cycle monitoring time

- Retentivity, .e. the number of flags, timers and counters retained

- Memory, such as local data

    Note: If you modify the distribution of RAM in your parameters, the CPU reorganizes all system data when it loads these to the RAM. The result is, that all DBs generated by means of SFC will be deleted, and the remaining data blocks will be initialized with the values from load memory.

    The RAM area available for code and data blocks will be modified if you edit the following parameters:

    – size of the process image, by bytes at the "Cycle/Clock memory"tab

    – Communication resources in the "Memory" tab

    – Size of the diagnostics buffer at the "Diagnostics/Clock" tab

    – Number of local data for all priority classes at the "Memory" tab.

- Assignment of interrupts (process interrupts, delay interrupts, asynchronous error interrupts) to the priority classes

- Time-of-day interrupts, such as start, interval duration, priority

- Watchdog interrupts, such as priority, interval length

- Diagnostics/Clock, such as time synchronization

- Protection levels

- H parameters

Engineering tool

You can set the individual CPU parameters using "Configuring Hardware" in STEP 7.

Note

When you modify the parameters listed below, the operating system initializes the following elements.

- Size of the process image of the inputs
- Size of the process image of the inputs
- Size of the local data
- Number of diagnostics buffer inputs
- Communication resources

These initializations are:

- Data blocks will be initialized with the load values
- M, C, T, I, Q will be deleted, Irrespective of the retentivity setting (0)
- DBs generated by means of SFC will be deleted
- Permanently configured dynamic connections will be shut down

The system starts up in the same way as with a cold restart.

Further parameters

- The rack number of the redundant CPU, i.e. 0 or 1

  Use the selector switch on the rear panel of the CPU to change the rack number.

- The operating mode of a redundant CPU, single or redundancy mode

  For information on how to change the operating mode of a redundant CPU, refer to the appendix B.

S7-400H in Profibus DP Mode

5

Chapter Overview

5.1 CPU 41x-H as PROFIBUS DP master

Introduction

This chapter describes how to use the CPU as DP master and configure it for direct data exchange.

Chapter Overview

In Section	Description	On Page
5.1.1	DP address areas of 41xH CPUs	5-3
5.1.2	CPU 41xH as Profibus DP master	5-3
5.1.3	Diagnostics of a 41xH CPU operating as PROFIBUS DP master	5-6

Further References

For details and information on engineering, PROFIBUS subnetting and diagnostics within the PROFIBUS subnet, refer to the *STEP 7* online help system.

Further information

For details and information on migrating from PROFIBUS DP to PROFIBUS DPV1, refer to the Internet URL:

http://www.ad.siemens.de/simatic-cs

under contribution number 7027576

5.1.1 DP address areas of 41xH

Address areas of 41xH CPUs

Table 5-1   41x CPUs, MPI/DP interface as PROFIBUS DP

Address Area	414-4H	417-4H
MPI interface as PROFIBUS DP, inputs and outputs (bytes) in each case	2048	2048
DP interface as PROFIBUS DP, inputs and outputs (bytes) in each case	6144	8192
Of those addresses you can configure up to x bytes for each I/O in the process image	0 to 8192	0 to 16384

DP diagnostics addresses occupy at least one byte for the DP master and each DP slave in the input address area. At these addresses, the DP standard diagnostics can be called for the relevant nodeby means of the LADDR parameter of SFC13, for example.. Define the DP diagnostics addresses when you configure the project data. If you do not specify any DP diagnostics addresses, *STEP 7* automatically assigns the addresses as DP diagnostics addresses in descending order, starting at the highest byte address.

The slaves of a DPV1 master are usually assigned two diagnostics addresses.

5.1.2 41xH CPU as PROFIBUS DP master

Requirements

Configure the relevant CPU interface as DP master, i.e. make the following settings in *STEP 7*:

- Assign a net

- Configure the CPU as DP master

- Assign a PROFIBUS address

- Select the operating mode, S7 compatible or DPV1

    Default setting is DPV1

- Interconnect the DP slaves with the DP master system

Note

Is one of the PROFIBUS DP slaves a 31x or 41x CPU?

If yes, you will find it in the PROFIBUS DP catalog as a "preconfigured" station. Assign this DP slave CPU a slave diagnostics address in the DP master. You must couple the DP master to the DP slave CPU, and specify the address areas for the transfer of data to the DP slave CPU.

Monitor/modify, programming via PROFIBUS

As an alternative to the MPI interface, you can use the PROFIBUS DP interface to program the CPU or execute the Monitor/Modify PG functions via the PROFIBUS DP interface.

Note

The applications "Programming" or "Monitor/Modify" prolong the DP cycle if executed via PROFIBUS DP interface.

Startup of the DP master system

Edit the following parameters to configure startup monitoring of the DP master:

- Ready message from module
- Parameter transfer to modules

In other words, the DP slaves must be started up and configured by the CPU (as DP master) within the set time.

PROFIBUS address of the DP master

All PROFIBUS addresses are permissible.

Step from IEC 61158 to DPV1

The enhancements in the IEC 61158 standard for distributed I/O were incorporated into IEC 61158 / IEC 61784-1:2002 Ed1 CP 3/1. The SIMATIC documentation refers to DPV1 in this context. The new version features various expansions and simplifications.

The DPV1 functions are integrated into the SIEMENS automation components. In order to be able to use these new features, you first have to lightly modify your system. A full description of the migration from IEC 61158 to DPV1 is available in the FAQ section titled "50170 Migrating from IEC 61158 to DPV1", FAQ contribution ID 7027576 on the Customer Support Internet pages.

Components Supporting PROFIBUS DPV1 functionality

DPV1 master

- The S7-400 CPUs with integrated DP interface.
- CP 443-5, order number 6GK7 443-5DX03-0XE0, 6GK7 443-5DX04-0XE0.

DPV1 slaves

- DP slaves listed the STEP 7 hardware catalog under their family names can be recognized in the information text as DPV1 slaves.
- DP slaves integrated in STEP 7 by means of GSD files revision 3 or higher.

Operating modes available for DPV1 components

- S7-compatible mode

  Components operated in a mode compatible to IEC 61158 do not provide full support of DPV1 functionality.

- DPV1 mode

  In this mode you have full access to DPV1 functionality. The station's automation components which do not support DPV1 can be used as before.

Compatibility between DPV1 and IEC 61158?

You may use all previous slaves after a conversion to DPV1. However, these slaves do not support the extended DPV1 functionality.

You can you use DPV1 slaves without a conversion to DPV1. The DPV1 slaves then behave like conventional slaves.. SIEMENS DPV1 slaves can be operated in S7–compatible mode. To integrate DPV1 slaves from other manufacturers, you need a GSD file to EN50170 earlier than revision 3.

Determining the bus topology in a DP master system using SFC103 "DP_TOPOL"

The diagnostics repeater is provided to improve the ability to locate disrupted modules or an interruption on the DP cables when failures occur in ongoing operation. This module is a slave that determines the topology of a DP strand and records any faults originating from it.

You can use SFC 103 "DP_TOPOL" to trigger the analysis of the bus topology of a DP master systems by the diagnostics repeater. SFC103 is described in the corresponding Online Help and in the*"System and Standard Functions* manual. For information on the diagnostics repeater refer to the *"Diagnostics Repeater for PROFIBUS DP*manual, order no. 6ES7972-0AB00-8BA0.

5.1.3 Diagnostics of a 41xH CPU operating as PROFIBUS DP master

Diagnostics Using LEDs

Table 5-2 shows the meaning of the BUSF LED.
The BUSF LED assigned to the interface configured as the PROFIBUS DP interface will always light up or flash.

Table 5-2 Meaning of the BUSF LEDs of the CPU 41x as DP master

BUSF	Meaning	What to Do
Off	Configuration correct All the configured slaves are addressable	–
is lit	• Bus error (hardware fault) • DP interface fault • Different transmission rates in multi–DP master operation (only in stand–alone mode)	• Check whether the bus cable has a short–circuit or a break. • Evaluate the diagnosis. Reconfigure or correct the configuration.
Flashes	• Station failure • At least one of the assigned slaves is not addressable	• Check whether the bus cable is connected to the CPU 41x or whether the bus is interrupted. • Wait until the CPU 41x has powered up. If the LED does not stop flashing, check the DP slaves or evaluate the diagnosis of the DP slaves.

Reading Out the Diagnostics Information with STEP 7

Table 5-3   Reading out the diagnostics information with *STEP 7*

DP Master	Block or Tab in *STEP 7*	Application	Refer To...
CPU 41x	DP slave diagnostics tab	To display the slave diagnosis as plain text at the *STEP 7* user interface	see "Hardware diagnostics" in the *STEP 7* Online Help, and the *Configuring hardware and connections with STEP 7 manual.*
	SFC 13 "DPNRM_DG"	Reading slave diagnostics data, i.e. saving these to the data area of the user program The busy bit may not be set to "0" when an error occurs while SFC13 is being processed. You should therefore check the RET_VAL parameter whenever SFC13 was processed.	For information on the structure for a 41x CPU, refer to the *CPU Data* reference manual; on the SFC, refer to the *System and Standard Functions* reference manual For information on the structure for other slaves, refer to the corresponding description.
	SFC 59 "RD_REC"	To read out data records of the S7 diagnosis (save these to the data area of the user program	
	SFC 51 "RDSYSST"	To read out SSL sublists Call SFC 51 in the diagnostics interrupt using the SSL ID W#16#00B3 and read out the SSL of the slave CPU.	**See the** *System and Standard Functions* Reference Manual
	SFB 52 "RDREC"	For DPV1 slaves: Reading data records of S7 diagnostics,i.e. saving these to the data area of the user program	
	SFB 54 "RALRM"	For DPV1 slaves: To read out interrupt information within the associated interrupt OB	

Evaluating diagnostics data in the user program

The figure below shows how to evaluate the diagnostics data in the user program.
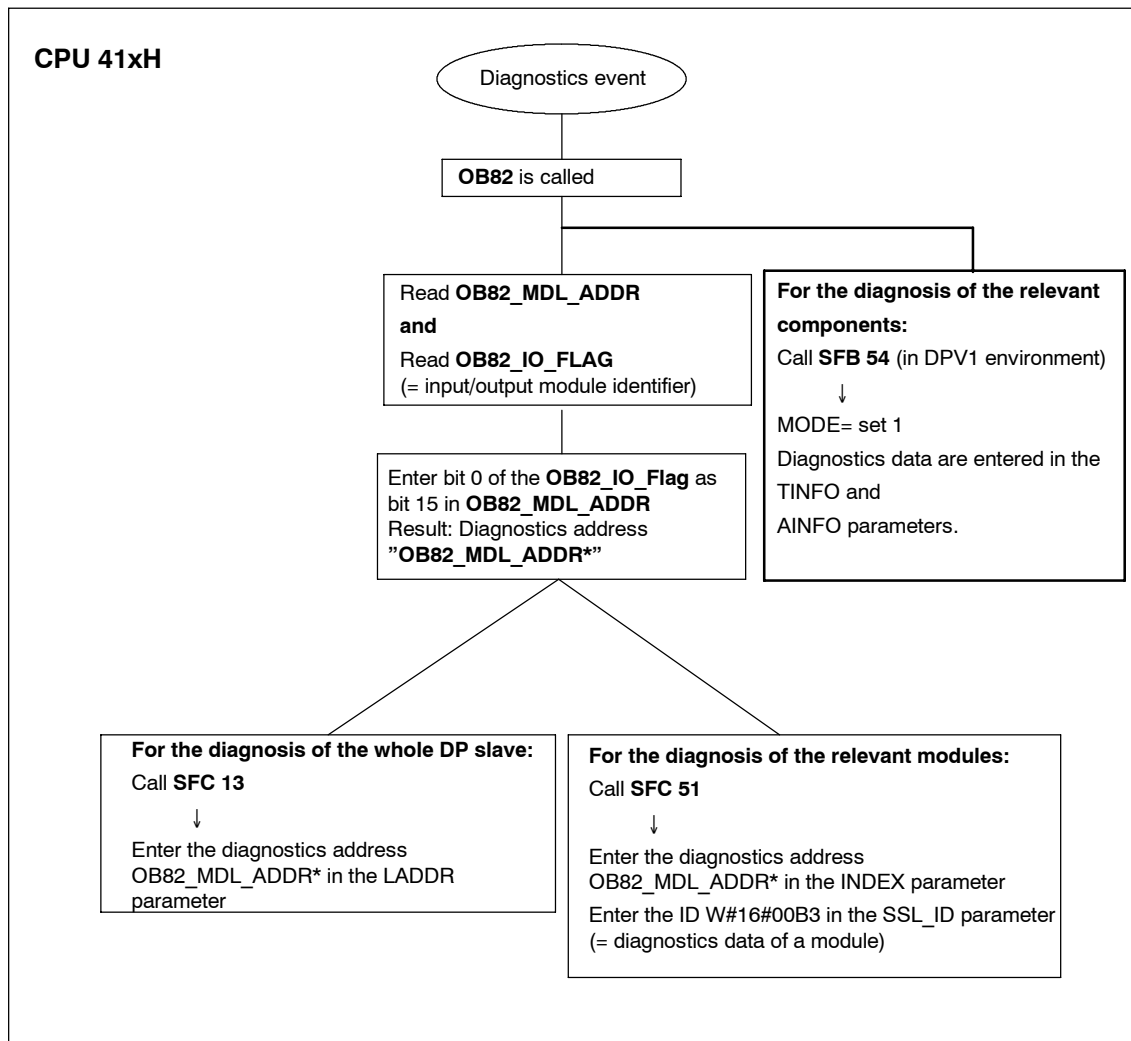
Figure 5-1   Diagnostics with CPU 41xH

Diagnostics Addresses in Connection with DP Slave Functionality

Assign the diagnostics addresses for PROFIBUS DP at the 41xH CPU. Ensure during configuration that DP diagnostics addresses are assigned once to the DP master and once to the DP slave.

S7-CPU as DP master

DP-Slave

PROFIBUS

You specify two diagnostics addresses during configuration:

Diagnostics address

Diagnostics address

During the configuration of the DP master, you specify (in the associated project of the DP master) a diagnostics address for the DP slave. In the following, this diagnostics address is described as being *assigned to the DP master*.

The DP master reads information on the status of the DP slave and any bus interruptions at this diagnostics address. See also table 5-4.

During the configuration of the DP slave, you also specify (in the associated project of the DP slave) a diagnostics address that is assigned to the DP slave. In the following, this diagnostics address is described as being *assigned to the DP slave*.

By means of this diagnostics address the DP slave receives information on the status of the DP master or a bus interruption.

Figure 5-2   Diagnostics addresses for the DP master and DP slave

Event detection

Table 5-4 shows how the 41xH CPU in DP master mode detects operating state transitions at a DP slave or interruptions of the data transfer.

Table 5-4   Event detection of 41xH CPUs in DP master mode

Event	What Happens in the DP Master
Bus interruption due to short–circuit or disconnection of the bus connector	• OB 86 called with the message *Station failure* incoming event; diagnostics address of the DP slave that is assigned to the DP master • with I/O access: call of OB 122, I/O access error
DP slave: RUN → STOP	• Call of OB 82 with the message *Module error* as coming event; Diagnostics address of the DP slave which is assigned to the DP master; tag OB82_MDL_STOP=1
DP slave: STOP → RUN	• Call of OB 82 with the message *Module OK* as going event; diagnostic address of the DP slave which is assigned to the DP master; tag OB82_MDL_STOP=0

Evaluation in the User Program

The table below shows you how to evaluate RUN-STOP transitions of the DP slave in the DP master. See also table 5-4.

In the DP Master	In the DP Slave (CPU 41x)

Example of diagnostics addresses:
    Master diagnostics address=**1023**
    Slave diagnostics address in the master system=**1022**
	Example of diagnostics addresses:
	    Slave diagnostics address =**422**
	    Master diagnostics address = irrelevant

The CPU calls OB 82 with the following information, amongst other things: • OB 82_MDL_ADDR:=**1022** • OB82_EV_CLASS:=B#16#39 as coming event • OB82_MDL_DEFECT:=module malfunction This information is also available in the diagnostics buffer of the CPU Your application program should also be set up for reading the diagnostics data of the DP slave using SFC 13 "DPNRM_DG".. In the DPV1 environment, use SFB54.It outputs the entire interrupt information.	←——— CPU: RUN → STOP CPU generates a DP slave diagnostics frame.

5.2 Consistent Data

Data that belongs together in terms of its content and a process state written at a specific point in time is known as consistent data.. In order to maintain data consistency, do not modify or update these during their transfer.

Example 1:

In order to provide a consistent image of the process signals to the CPU for the duration of cyclic program execution, the process signals are written to the process image of inputs prior to program execution, or the processing results are written to the process image of outputs after program execution. Subsequently, during program scanning when the address area "inputs" (I) and "outputs" (O) are addressed, the user program addresses the internal memory area of the CPU on which the image of the inputs and outputs is located instead of directly accessing the signal modules.

Example 2:

Inconsistency may develop when a communicationblock, such as SFB 14 "GET" or SFB 15 "PUT" is interrupted by a process alarm OB of higher priority. When the user program modifies any data of this process alarm OB which have already been processed by the communication block, -certain parts of the transferred data will have retained their original status which was valid prior to process alarm processing, while others represent delta data as a result of process alarm processing.

This results in inconsistent data, i.e. data which are no longer associated.

SFC 81 "UBLKMOV"

Use SFC 81 "UBLKMOV" to copy the content of a memory area of the source consistently to another memory area, namely the destination area. The copy operation can not be interrupted by other operating system activities.

SFC 81 "UBLKMOV" enables you to copy the following memory areas:

- Memory markers
- DB contents
- Process image of the inputs
- Process image of outputs

The maximum amount of data you can copy is 512 bytes. Make allowances for certain CPU-specific restrictions listed in the instructions list.

Since copying can not be interrupted, the interrupt reaction times of your CPU may increase when using SFC 81 "UBLKMOV".

The source and destination areas must not overlap. If the specified destination area is larger than the source area, the function only copies as much data to the destination area as that contained in the source area. If the specified destination area is smaller than the source area, the function only copies as much data as can be written to the destination area.

5.2.1 Consistency of communication blocks and functions

Using S7–400 the communication data is not processed in the scan cycle checkpoint; instead, this data is processed in fixed time slices during the program cycle.

The system can always process the data formats byte, word and dword consistently, i.e. the transfer or processing of 1 byte, 1 word = 2 bytese or 1 dword= 4 bytes can not be interrupted.

When the user program calls communication blockssuch as SFB 12 BSEND" and SFB 13 BRCV", which are only used in pairs and access shared data, the access to this data area can be coordinated by means of the actual "DONE" parameter, for example. Data consistency of the communication areas transmitted locally with a communication block can thus be ensured in the user program.

In contrast, S7 communication functions do not require a block, such as SFB 14 "GET", SFB 15 "PUT",in the user program of the PLC.Here, you should make allowances for the volume of consistent data in the programming phase.

5.2.2 Access to the Working Memory of the CPU

The communication functions of the operating system access the working memory of the CPU in fixed block lengths. The block length is CPU-specific. The tags for S7-400 CPUs have a length of up to 472 bytes.

This ensures that the interrupt reaction time is not due to communication load. Since this access is performed asynchronously to the user program, you can not transmit an unlimited number bytes of consistent data.

The rules to ensure data consistency are described below.

5.2.3 Consistency rules for SFB 14 "GET" or reading tag and SFB 15 "PUT" or writing tag

SFB 14

The data are received consistently if you observe the following points:

Evaluate the entire, currently used part of the receive area RD_i before you activate a new request.

SFB 15

When you initiate a send operation (positive edge at REQ), the system copies the data of the send data areas SD_i to be transferred from the user program. You can write new data to these areas after the block call, without any risk of corrupting the current send data.

Note

The transfer operation is not completed until the status parameter DONE assumes the value 1.

5.2.4 Reading Data consistently from a DP Standard Slave and Writing Consistently to a DP Standard Slave

Reading Data Consistently from a DP Standard Slave Using SFC 14 "DPRD_DAT"

Use SFC 14 "DPRD_DAT" "read consistent data of a DP standard slave" to consistently read the data of a DP standard slave.

The data read is entered into the destination range defined by RECORD if no error occurs during the data transmission.

The destination range must have the same length as the one you have configured for the selected module with STEP 7.

By invoking SFC 14 you can only access the data of one module / DP ID at the configured start address.

Writing Data Consistently to a DP Standard Slave Using SFC 15 "DPWR_DAT"

Use SFC 15 "DPWR_DAT", "write consistent data to a DP standard slave" to transfer the data in RECORD consistently to the addressed DP standard slave.

The source range must have the same length as the one you have configured for the selected module with STEP 7.

Note

The PROFIBUS DP standard defines high limits for the transfer of consistent user data, see the next chapter. Typical DP standard slaves adhere to those high limits. CPUs released prior to 1999 have CPU–specific restrictions with respect to the transfer of consistent user data. For these CPUs you can determine the maximum length of the data which the CPU can consistently read and write to and from the DP standard in the respective technical specifications under the index entry "DP Master – User data per DP slave". Newer CPUs are capable of exceeding the value for the amount of data that a DP standard slave can send and receive.

Upper Limit for the Transmission of Consistent User Data on a DP Slave

The PROFIBUS DP standard defines the upper limit for the transmission of consistent user data to a DP slave.

For this reason a maximum of 64 words = 128 bytes of user data can be consistently transferred in a block to the DP slave.

You can define the length of consistent area in your configuration. In the special identification format (SIF), you can define a maximum length of consistent data of 64 words = 128 bytes, 128 bytes for inputs, and 128 bytes for outputs. Any greater length value is not possible.

This high limit applies only to pure user data. Diagnostics and parameter data will be grouped to form complete data records, and are thus always transferred consistently.

In the general identification format (GIF), you can define a maximum length of consistent data of 16 words = 32 bytes, 32 bytes for inputs, and 32 bytes for outputs.Any greater length value is not possible.

In this context, make allowances for the fact that a 41x CPU operating as DP slave generally has to support its configuration at an external master (implementation by means of GSD file) using the general identification format. A 41x CPU operated as DP slave thus supports only a maximum length of 16 words = 32 bytes in its transfer memory for PROFIBUS DP.

5.2.5 Consistent Data Access without the Use of SFC 14 or SFC 15

Consistent data access > 4 bytes is also possible without using SFC14 or SFC15. The data area of a DP slave which is to be transferred consistently will be written to a process image partition. The data in this area are thus always consistent. You can then access the process image partition using the load / transfer commands (L EW 1, for example). This represents a particularly comfortable and efficient (low runtime load) method to access consistent data and to implement and configure such devices as drives or other DP slaves.

Any direct access to a data area which is configured consistent, such as L PEW or T PAW, does **not** result in an I/O access error.

Important aspects in the conversion from the SFC14/15 solution to the process image solution are:

*   When converting from the SFC14/15 method to the process image method, it is not recommended to use the system functions and the process image at the same time. Although the process image is updated when writing with the system function SFC15, this is not the case when reading. In other words, consistency between the values of the process image and of the system function SFC14 is not ensured.

*   SFC 50 "RD_LGADR" outputs another address area with the SFC 14/15 method as with the process image method.

*   When using a CP 443-5 ext, the parallel use of system functions and of the process image leads to the following errors:read/write access to the process image is blocked, and/or SFC 14/15 is no longer able to perform any read/write access operations.

Example:

The example of the process image partition 3 "TPA 3"below shows a possible configuration in HW Config:

- TPA 3 at output: those 50 bytes are stored consistently in process image partition 3 (pull–down list "Consistent over –> entire length"), and can thus be read by means of standard "load input xy" commands.

- Selecting "Process Image Partition –> –––" under input in the pull–down menu means: do not write data to the process image. You must work with the system functions SFC14/15. .

System and Operating Modes of the S7-400H

6

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

You will learn the basic concepts that are used in describing how redundant systems operate.

Following that, you will receive information on redundant system modes. These modes depend on the operating modes of the different redundant 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 redundant CPUs.

In Section	Description	On Page
6.1	Introduction	6-2
6.2	System Modes of the S7-400H	6-5
6.3	Operating Modes of the CPUs	6-6
6.4	Self-test	6-12
6.5	Time Response	6-16
6.6	Evaluation of process alarms in the S7-400H System	6-16

6.1   Introduction

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

The two subsystems create a redundant 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, commonly also referred to as functional redundancy, means that all redundant resources are constantly in operation and simultaneously involved in the execution of the control task.

To the S7-400H this implies that the user programs in both CPUs are identical and executed synchronously by the CPUs.

Agreement

To distinguish between both units, we use the traditional expressions of "master" and "standby" for dual-channel redundant systems in this description. The standby always processes events in synchronism with the master, and does not explicitly wait for any errors before doing so.

The distinction made between the master and standby CPUs is primarily important for ensuring reproducible error reactions. Hence, the standby CPU may go into STOP when the redundant coupling fails, while the master CPU remains in RUN.

Master/standby assignment

When the S7-400H is initially switched on, the first CPU to be started assumes master mode, and the partner CPU assumes standby mode.

This master/standby setting is then retained when both CPUs simultaneously POWER ON.

This master/standby setting changes when:

1. The standby CPU starts up before the master CPU (interval of at least 3 s)

2. The redundant master CPU fails or goes into STOP

3. No fault was found in TROUBLESHOOTING mode (refer also to Section 6.3.6)

Synchronizing the units

The master and standby CPUs are coupled by means of fiber-optic cables. The redundant CPUs maintain event-driven synchronous program execution via this coupling.

```
┌─────────────────────────────────────────────────────────────┐
│                                                             │
│  ┌──────────────┐                        ┌──────────────┐   │
│  │              │                        │              │   │
│  │  Subsystem   │  ◄─── Synchronization ───►  │  Subsystem   │   │
│  │              │                        │              │   │
│  │  (CPU0)      │                        │  (CPU1)      │   │
│  │              │                        │              │   │
│  └──────────────┘                        └──────────────┘   │
│                                                             │
└─────────────────────────────────────────────────────────────┘
```
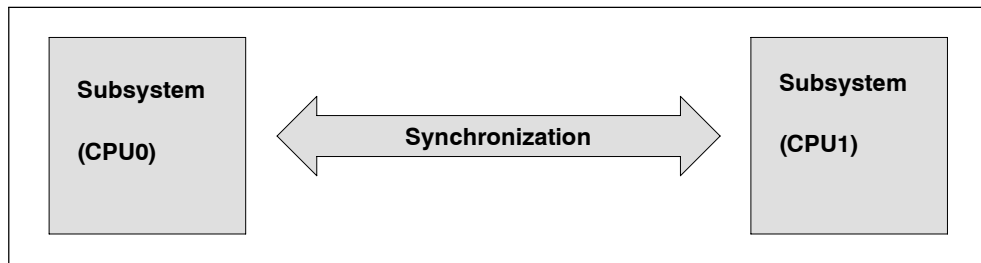
Figure 6-1   Synchronizing the subsystems

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 the master and standby units always synchronize their data when an event occurs which may lead to different internal states of the units.

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

Bumpless continuation of operation in case of redundancy loss at a CPU

The event-driven synchronization method ensures bumpless continuation of operations by the standby CPU even in the event of a master failure.

Self-test

Malfunctions have to be detected, isolated and reported as quickly as possible. Consequently, wide-ranging self-test functions have been implemented in the S7-400H that run automatically and entirely in the background.

The following components and functions are tested:

- interconnection of the CPUs

- processor

- Internal memory of the CPU

- I/O bus

If the self-test detects an error, the redundant system tries to eliminate it or to suppress its effects.

For detailed information on the self–test, refer to chapter 6.4.

6.2 States of the S7-400H system

The system status of the S7-400H is derived from the operating states of the two CPUs. The term "system status" is used in order to obtain a simplified expression which identifies the concurrent operating states of the two CPUs.

Example: Instead of "the master CPU is in RUN and the standby CPU is in COUPLING mode" we use "the S7-400H system is in coupling mode".

Overview of the system states

The table below provides an overview of the various states of the S7-400H system.

Table 6-1   Overview of the S7-400H system states

System states of the S7-400H	Operating states of the two CPUs	
	Master	**Standby**
Stop	STOP	STOP, power off, DEFECTIVE
Startup	STARTUP	STOP, power off, DEFECTIVE, no synchronization
Stand–alone mode	RUN	STOP, TROUBLESHOOTING, power off, DEFECTIVE, no synchronization
Coupling	RUN	STARTUP, COUPLING
Update	RUN	UPDATE
Redundant mode	RUN	RUN
Hold	HOLD	STOP, TROUBLESHOOTING, power off, DEFECTIVE, no synchronization

6.3 Operating states of the CPUs

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

Operating states from POWER ON to system redundancy

Generally speaking, the two CPUs enjoy equal rights so that either can be the master or the standby CPU. For reasons of legibility, the illustration presumes that the master CPU (CPU 0) is started up before the standby CPU (CPU 1) was switched on.

Figure 6-2 deals with the operating states of the two CPUs, from POWER ON up to system redundancy. The HOLD (refer to chapter 6.3.5) and TROUBLESHOOTING (refer to chapter 6.3.6) operating states are not included, because these take a special position.
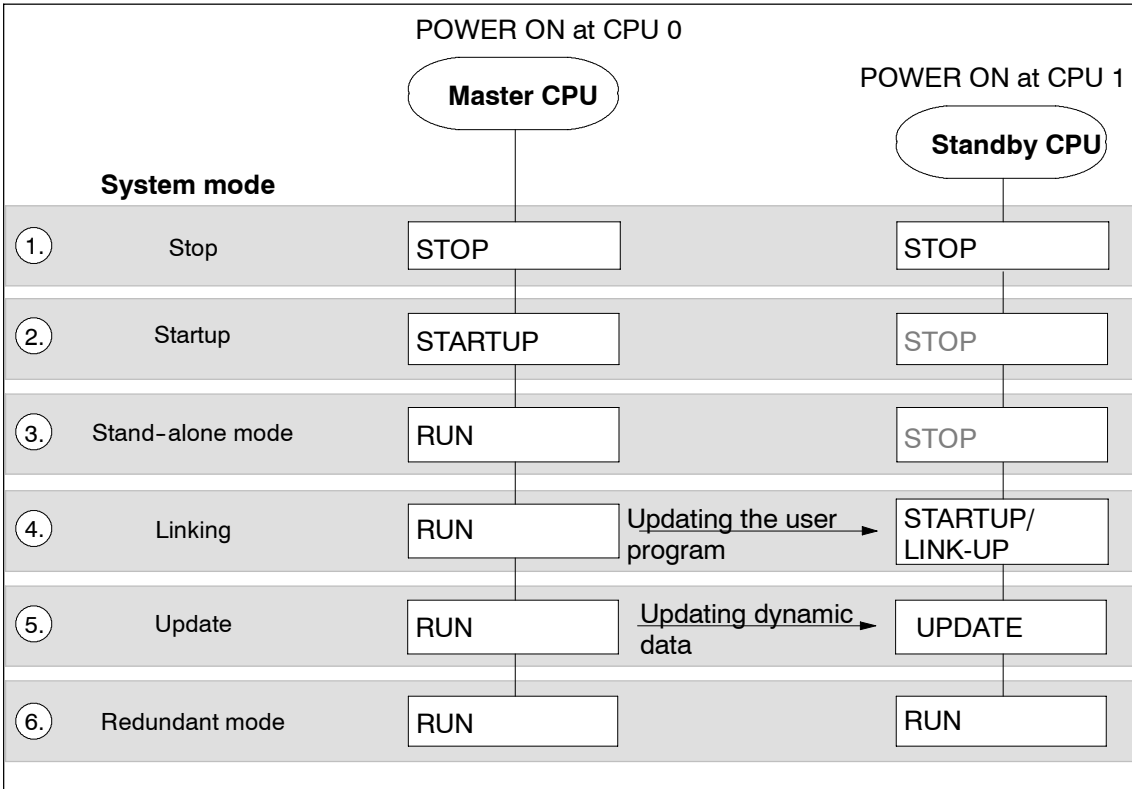
Figure 6-2   System and operating modes of the redundant system

Explanations relating to Figure 6-2

Table 6-2   Explanations relating to figure 6-2 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 into STARTUP state and executes OB100 or OB102 according to the startup mode, see also chapter 6.3.2.
3.	If startup is successful, the master CPU (CPU 0) changes to stand–alone mode. Now, only themaster CPU now executes the user program. At the transition to the COUPLING system state, no block may be opened by the "Monitor" option, and no tag table may be active.
4.	If the standby CPU (CPU 1) requests COUPLING, the master and standby CPUs compare their user programs. If any differences are found, the master CPU updates the user program of the standby CPU, see also chapter 6.3.3.
5.	The update starts when coupling is completed, see also chapter 7.3.2. The master CPU updates the dynamic data of the standby CPU in this step. Dynamic data are inputs, outputs, timers, counters, flags and data blocks.. The memory content is thus identical in both CPUs as a result of the update, see also chapter 6.3.3.
6.	The master and standby CPUs are in RUN after the update and process the user program in synchronism. Exception: master/standby changeover for configuration/program modification. redundant state is only possible if both CPUs are of the same release and the same firmware version.

6.3.1   STOP operating state

Except for the additions described below, the behavior of S7-400H CPUs in STOP corresponds with that of standard S7-400 CPUs.

When you download a configuration to one of the CPUs while both are in STOP, make allowances for the points outlined below:

*   Start the CPU to which you downloaded the configuration first, in order to set it up for master mode.

*   By initiating the system startup request on the PG, you first start the CPU to which an online connection exists,regardless of the master or standby status.

Notice

A system startup may trigger a master-standby changeover.

CPU memory reset

The CPU memory reset function affects only the selected CPU. To reset both CPUs, you must do so successively.

6.3.2 STARTUP operating state

Except for the additions described below, the STARTUP behavior of S7-400H CPUs corresponds with that of standard S7-400 CPUs.

Startup

The redundant CPUs distinguish between the cold restart and restart (warm restart).

Fault-tolerant CPUs do not support hot restarts.

Startup processing of the master CPU

The startup system status of an S7-400H is always processed by the master CPU..

During STARTUP, the master CPU compares the online I/O configuration with the hardware configuration that you created offline in STEP 7. If any differences are found, the master CPU reacts in the same way as a standard S7-400 CPU.

The master CPU checks and configures:

- the switched I/O
- its assigned one–sided I/O

Startup of the standby CPU

The standby startup routine does not call an OB100 or OB102.

The standby CPU checks and configures:

- its assigned one–sided I/O

Further information

For detailed information on STARTUP states, refer to the *Programming with STEP 7* manual.

6.3.3 COUPLING and UPDATE operating states

The master CPU checks and updates the memory contents of the standby CPU before the redundant system assumes redundant mode. This action involves two successive phases, namely the coupling and update phases.

The master CPU is always in RUN and the standby CPU is in COUPLING or UPDATE status during the coupling and update phases.

In addition to the coupling and update functions which are carried out in order to establish system redundancy, the system also supports coupling and updating in combination with master/standby changeover.

For detailed information on coupling and updating, refer to chapter 7.

6.3.4 Operating State RUN

Except for the additions described below, the RUN behavior of S7-400H CPUs corresponds with that of standard S7-400 CPUs.

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

- Stand–alone mode
- Coupling, updating
- Redundant mode

Stand–alone mode, coupling, updating

In the system states mentioned above, the master CPU is in RUN and executes the user program in stand–alone mode.

Redundant system Mode

The master and standby CPUs are always in RUN when operating in redundant state, execute the user program in synchronism, and perform mutual checks.

In redundant state, it is not possible to test the user program with breakpoints.

Redundant state is only supported with CPUs of the same release and firmware version. Redundancy will be lost if one of the error listed in Table 6-3 occurs.

Table 6-3   Causes of error leading to redundancy loss

Cause of error	Reaction
Failure of one CPU	Refer to chapter 11.1.1
Failure of the redundancy coupling (synchronization module or fiber-optic cable)	Refer to chapter 11.1.5
RAM comparison error	See chapter 6.3.6.

Redundant use of modules

The following rule applies to redundant state:

Modules interconnected in redundant mode, such as DP slave interface module IM 153-2, must have the same order number, version and firmware version.

6.3.5   HOLD operating state

Except for the additions described below, the HOLD behavior of S7-400H CPUs corresponds with that of standard S7-400 CPUs.

The HOLD state takes an exceptional position, for it is used only for test purposes.

Prerequisite of the HOLD state

A transition to HOLD is only available during STARTUP and in RUN of the stand–alone unit.

Characteristics

- Coupling and update operations are not available while the redundant CPU is in HOLD state, and the standby CPU remains in STOP and outputs a diagnostics message.

- It is not possible to set breakpoints if the redundant system remains in redundant state.

6.3.6 TROUBLESHOOTING operating state

The TROUBLESHOOTING state is only available in redundant system state.

The self-test routine compares the master and standby CPUs, and reports an error if any differences are found. Errors could be caused by hardware faults, checksum errors and RAM/PIO comparison errors.

The following events will trigger the TROUBLESHOOTING state:

1. If a one-sided call of OB121 (only on one CPU) is output in redundant mode, the CPU assumes a hardware fault and enters the TROUBLESHOOTING state. The partner CPU assumes master mode, and continues operation in stand–alone mode if required.

2. When a checksum error occurs on only one of the redundant CPUs, this CPU will enter the TROUBLESHOOTING state. The partner CPU assumes master mode, and continues operation in stand–alone mode if required.

3. When a RAM/PIO comparison error is detected in redundant mode, the standby CPU enters the TROUBLESHOOTING state (defaultreaction), and the master CPU continues operation in stand–alone mode.

   The reaction to RAM/PIQ comparison errors can be modified in the configuration (for example, the standby CPU goes into STOP).

4. When a multpile–bit error occurs on only one of the redundant CPUs, this CPU will enter the TROUBLESHOOTING state. The partner CPU assumes master mode, and continues operation in stand–alone mode if required.

   However: OB84 is called when a single–bit error occurs on one of the redundant CPUs, and the CPU will not enters the TROUBLESHOOTING state.

The TROUBLESHOOTING state is set in order to localize a faulty. The standby CPU executes the entire self-test, while the master CPU remains in RUN.

When a hardware error is detected, the CPU enters the
DEFECTIVE status, otherwise the CPU is coupled again.
The redundant system recovers redundant state, and an automatic master-standby changeover will be carried out. This ensures that when the next error is detected in troubleshooting mode, the hardware of the previous master CPU is tested.

The CPU inTROUBLESHOOTING state does not allow online access, for example, by the PG. The TROUBLESHOOTING state is indicated at the RUN and STOP LEDs. See chapter 4.3.

For further information on the self-test, refer to chapter 6.4.

6.4   Self-test

Processing self-tests

The CPU executes the entire self–test program after POWER ON without backup voltage, for example, POWER ON after initial insertion of the CPU, or POWER ON without backup battery, and in TROUBLESHOOTING state. The self-test takes approx.10 minutes.

When the CPU of a redundant system request a CPU memory reset and is then shut down with backup power, it performs a self–test irrespective of the backup function. The CPU requests a memory reset when you remove the Memory Card, for example.

The runtime operating system splits the self–test routine into several small program sections, namely the test slices,which are processed in multiple successive cycles. The cyclic self–test is organized to perform a single, complete pass. The default time of 90 minutes can be modified in the configuration.

Reaction to errors during the self-test

If the self–test returns an error, the following happens:

Table 6-4   Reaction to errors during the self-test

Error class	System reaction
Hardware error without one–sided call of OB 121	The faulty CPU enters the DEFECTIVE state. The redundant system changes over to stand–alone mode. The cause of the error will be written to the diagnostics buffer.
Hardware error with one-sided call of OB121I	The CPU with the one-sided OB121 enters the TROUBLESHOOTING state. The redundant system changes over to stand–alone mode (see below).
RAM/PIQ comparison error	The cause of the error will be written to the diagnostics buffer. The CPU enters the configured system or operating state (see below).
Checksum error	The reaction depends on the error situation (see below).
Multiple–bit error	The faulty CPU enters TROUBLESHOOTING state.

Hardware error with one-sided call of OB121I

If a hardware error occurs with a one-sided OB121 call for the first time since the previous unbuffered POWER ON, the faulty CPU enters the TROUBLESHOOTING state. The redundant system changes over to stand–alone mode, and the cause of the error will be written to the diagnostics buffer.

RAM/PIO comparison error

If the self-test returns a RAM/PIQ comparison error, the redundant system quits redundant mode and the standby CPU partners the TROUBLESHOOTING state (default configuration). The cause of the error will be written to the diagnostics buffer.

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

Table 6-5   Reaction to a recurring comparison error

Comparison error persists ...	Reaction
in the first self-test cycle after troubleshooting	The standby CPU first enters the TROUBLESHOOTING state, and then goes into STOP.
	The redundant system changes over to stand–alone mode.
after two or more self-test cycles after troubleshooting	Standby CPU enters the TROUBLESHOOTING state.
	The redundant system changes over to stand–alone mode.

Checksum errors

The system reacts to initial checksum error detected after the last unbuffered POWER ON as follows:

Table 6-6   Reaction to checksum errors

Time of detection	System reaction
During the POST	The faulty CPU enters the DEFECTIVE state.
	The redundant system changes over to stand–alone mode.
In the cyclic self-test (STOP or stand–alone mode)	The error will be rectified. The CPU remains in STOP or in stand–alone mode.
In the cyclic self-test (redundant state)	The error will be rectified. The faulty CPU enters TROUBLESHOOTING state.
	The redundant system changes over to stand–alone mode.
In TROUBLESHOOTING state	The faulty CPU enters the DEFECTIVE state.
Single–bit errors	The CPU calls OB84 after the detection and elimination of the error.

The cause of the error will be written to the diagnostics buffer.

In an F system, the F program is informed that the self-test has detected an error the first time a checksum error occurs in STOP or stand–alone mode. The reaction of the F program to this is described in the *S7-400F and S7-400FH Programmable Controllers* manual.

Hardware error with one−sided call of OB121, checksum error, second occurrence

The 41x-4H CPU reacts to a second occurrence of a hardware error with one−sided call of OB121 and of a checksum error as described in the table below, based on the various operating modes of the 41x-4H CPU.

Table 6-7   Hardware error with one−sided call of OB121, checksum error, second occurrence

Error	CPU in stand−alone mode	CPU in stand−alone mode	CPU in redundant mode
Hardware error with one-sided call of OB121	OB121 is being executed	OB121 is being executed	The faulty CPU enters TROUBLESHOOTING state. The redundant system changes over to stand−alone mode,
Checksum error	The CPU enters the DEFECTIVE status if two errors occur within two successive test cycles. (Configure the length of the test cycle in HW Config)	The CPU enters the DEFECTIVE status if two errors occur within two successive test cycles. (Configure the length of the test cycle in HW Config)	The CPU enters the DEFECTIVE status if a second error occurs within the troubleshooting state which was triggered by the first error event.

The CPU when operating on stand−alone or stand−alone mode reacts to a second checksum error as it did on the first occurrence of the error, after twice the test cycle time has expired. A CPU operating in redundant mode reacts to a second error (hardware error with one−sided call of OB121, checksum error) as it did on the first occurrence of the error and when troubleshooting is concluded.

Multiple-bit errors

The CPU enters the TROUBLESHOOTING state when a multiple−bit error is detected while the redundant system is operating in redundant mode. When troubleshooting is concluded, the CPU can automatically couple and update itself in order to resume redundant operation. At the transition to troubleshooting mode, the address of the triggering error is reported in the diagnostics buffer.

Single-bit errors

The CPU calls OB84 after the detection and elimination of the error.

Controlling the cyclic self–test 6-15

SFC90 H_CTRL allows you to control the scope and execution of the cyclic self-test. For example, you can remove various test components from this scope, or and include these again. In addition, you may explicitly call specific test components, and then initiate processing of these.

For detailed information on SFC90 H_CTRL, refer to the *System Software for S7-300/400, System and Standard Functions* manual.

Notice

In a fail-safe system, you may not disable and then re–enable the cyclic self-tests. For more details, refer to the *S7-400F and S7-400FH Programmable Controllers* manual.

6.5 Time–based reaction

Instruction run times

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

Processing I/O direct access

Please note that I/O accesses always require a synchronization of the two units, and thus extend the cycle time.

You should thus avoid any direct I/O access in your user program, and rather access the data using the process images (or the process image partitions, for example, when handling watchdog interrupts). This automatically increases performance, because at the process image you can always synchronize a value record in a single pass.

Reaction time

For detailed information on calculating reaction times, refer to Section 7.4.1.

Note that any update of the standby CPU extends the interrupt reaction time.

The interrupt reaction time depends on the priority class, because a graduated delay of the interrupts is performed during an update.

6.6 Evaluation of process alarms in the S7-400H System

The use of process alarm–triggering modules in the S7-400H system may cause inconsistency in the process values which can be read from the process alarm OB by means of direct access and the process values valid at the time of the interrupt. Hence, you should rather evaluate the temporary tags (start information) in the process alarm OB.

When using the process alarm–triggering module SM 321-7BH00, it is thus not advisable to have different reactions to positive or negative edges at the same input, because this would require direct access to the I/O. If you want to react differently to those signal transitions in your user program, assign the signal to two inputs at different channel groups, and configure one input for the positive, and the other for the negative edge.

Coupling and synchronization

7

7.1 Effect of coupling and update operations

Coupling and update operations are indicated by the REDF LEDs on the two CPUs. During coupling operations, those LEDs flash at a frequency of 0.5 Hz, and on update operations at a frequency of 2 Hz.
Coupling and update operations have various effects on user program execution and on communication functions.

Table 7-1   Properties of coupling and update functions

Process	Coupling	Update
Execution of the user program	All priority classes (OBs) are processed.	Processing of the priority classes is delayed by sections. All the requirements are caught up with after the update. For details, refer to the sections below.
Deletion, loading, generating and compressing blocks	Blocks can not be deleted, loaded, generated or compressed. If any of these actions are busy, coupling and update operations are inhibited.	Blocks can not be deleted, loaded, created or compressed.
Execution of communication functions, PG operation	Communication functions are being executed.	Execution of the functions is restricted by sections and delayed. All the delayed functions are caught up with after the update. For details, refer to the sections below.
CPU self-test	Not performed	Not performed
Test and commissioning functions, such as "Monitor and Control Tag", "Monitor (On/Off)"	Test and commissioning functions are disabled. When such actions are busy, coupling and update operations are inhibited.	Test and commissioning functions are disabled.
Handling of the connections to the master CPU	All connections are retained; no new connections can be made.	All connections are retained; no new connections can be made. Cancelled connections will not be established again until the update is completed
Handling of the connections to the standby CPU	All the connections will be cancelled; no new connections can be made.	All connections are down. These were cancelled during the coupling operation.

7.2 Conditions of coupling and updates

Which commands you may use on the PG to initiate a coupling and update operation is determined by the conditions at the master and standby CPU. The table below shows the correlation between those conditions and available PG commands for coupling and update operations.

Table 7-2   Conditions for coupling and update operations

Coupling and update operation as PG command:	Size and type of load memory in the master and standby CPUs	FW version in the master and standby CPUs	Available synchroni-zation couplings	HW version in the master and standby CPUs
Restart of the standby	are identical	are identical	2	are identical
Switch to CPU with modified configuration	RAM and EPROM mixed	are identical	2	are identical
Changeover to CPU with expanded memory	Size of load memory in the standby CPU is larger than that of the master	are identical	2	are identical
Changeover to the CPU with modified operating system	are identical	are different	2	are identical
CPUs with different hardware versions	are identical	are identical	2	are different
Only one synchronization coupling is available	are identical	are identical	1	are identical

7.3   Coupling and update operation

There are two types of coupling and update operations:

- Within a "normal" coupling and update operation, the redundant system should change over from stand–alone mode to **redundant** mode. The two CPUs will then process the same program in synchronism.

- After a coupling and update operation with **master/standby changeover**, the CPU containing the modified components can assume control over the process. Either the hardware configuration, or the memory configuration, or the operating system may have been modified.

   In order to return to redundant state, a "normal" coupling and update operation must be performed subsequently.

How to initiate the coupling and update operation?

Initial situation: stand–alone mode, i.e. only one of the CPUs of a redundant system connected via fiber-optic cables is in RUN.

To establish system redundancy, initiate the coupling and update operation as follows:

- Toggle the mode selector switch of the standby CPU from STOP to RUN.

- POWER ON the standby (mode selector switch in RUN position), if prior to POWER DOWN the CPU was not in STOP mode

- Operator input at the PG/ES.

A coupling and update operation with master/standby changeover is always initiated on the PG/ES.

Notice

The interruption of a coupling and update operation on the standby CPU (for example, due to POWER OFF, STOP) may cause data inconsistency, and thus lead to a request of the CPU to reset its memory.

The coupling and update functions are enabled again after a memory reset on the standby CPU.

Flow chart of the coupling and update operation

The diagram below outlines the sequence of a coupling and update operation in general terms. In the initial situation, the master is operating in stand–alone mode, shown as CPU 0 in the diagram.

Master CPU (CPU 0)   **Standby CPU** (CPU 1))

| RUN | **Coupling**
(REDF LEDs flash at 0.5 Hz) | STOP |

The standby requests COUPLING

The functions for deleting, loading, generating and compressing blocks are disabled.
Test and commissioning functions are disabled.

The functions for deleting, loading, generating and compressing blocks are disabled.
Test and commissioning functions are disabled.

Comparison of the memory configuration, operating system version and FEPROM contents

Copy load memory content *)
Copy user program blocks of main memory *)

All connections will be cancelled

Integration of the DP slaves

Integration of the connections

Update
see figure 7-2

Lift restrictions; catch up delayed execution

Lift restrictions; catch up delayed execution

Redundant system status, or master/standby changeover with new standby in STOP

*) When the "Switch to CPU with Modified Configuration" option is set, the contents of load memory will not be copied. For information on which data Will be copied from the user program blocks in the RAM (OBs, FCs, FBs, DBs, SDBs) area of the master CPU, refer to chapter 7.3.3.

Figure 7-1   Sequence of coupling and update operations
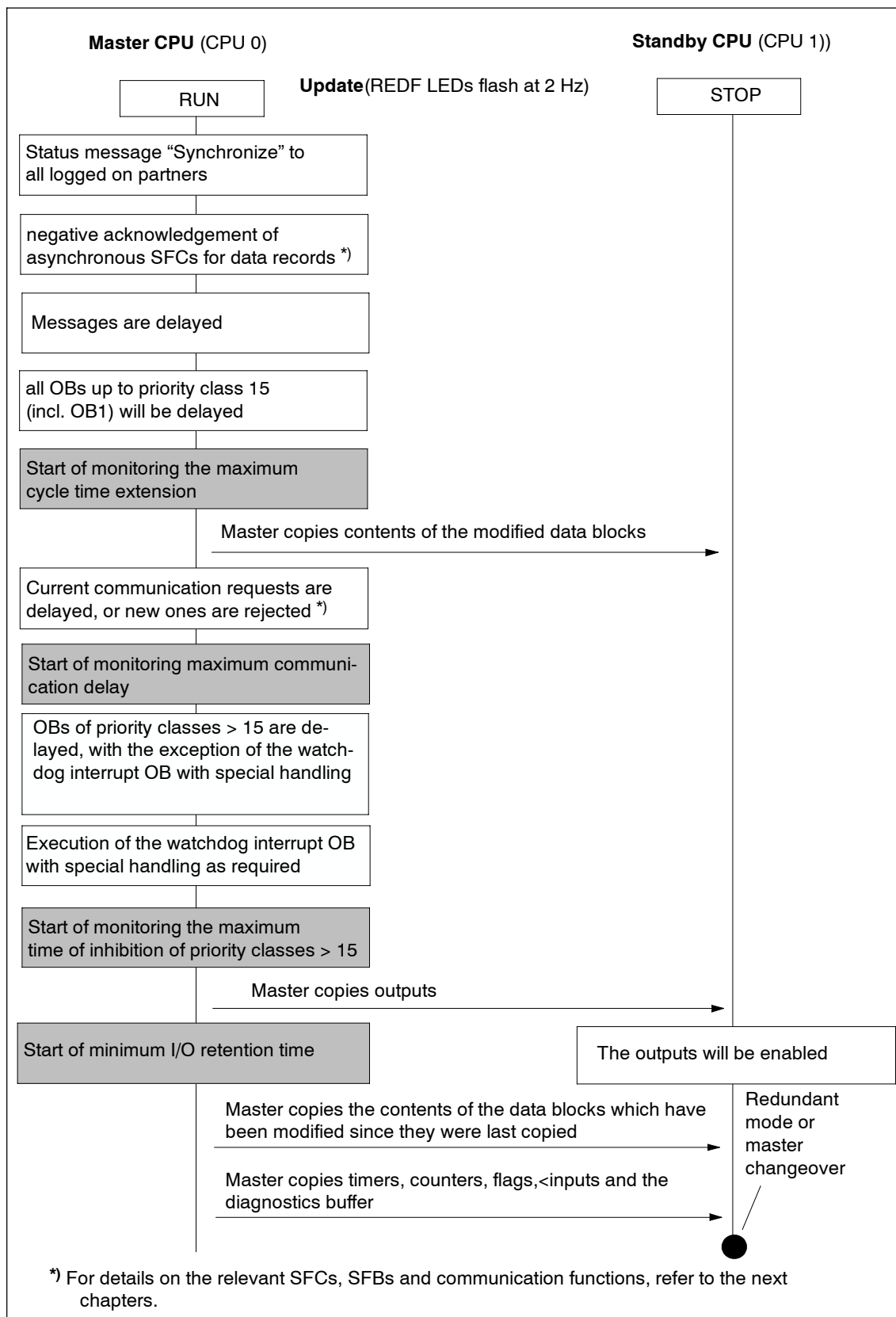
Figure 7-2   Sequence of update operations

Minimum length of input signals during update operations

Program execution is stopped for a certain time during the update (details are provided in the next chapters). In order to ensure that the CPU can reliably detect the transition of input signals during update operations, the following conditions must be satisfied:

Min. signal duration > 2 ⊕ the time required for I/O update (DP only)
                + call interval of the priority class
                + program execution time of the priority class
                + time required for the update
                + program execution time of higher-priority classes

Example:

Minimum signal duration of an input signal which is evaluated in a priority class > 15 (for example, OB 40).
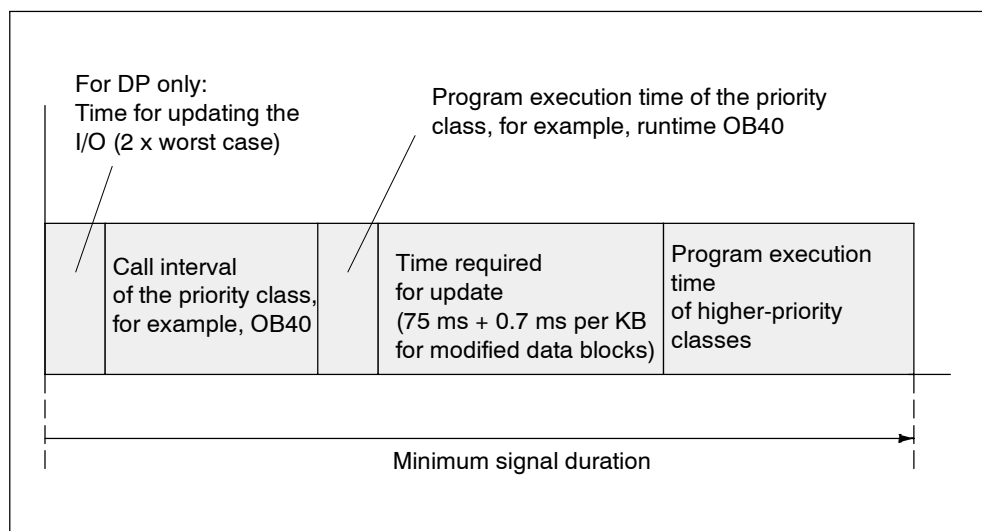
Figure 7-3   Example of minimum signal duration at an input signal during the update

7.3.1   Coupling sequence

For the coupling operation, you need to decide whether to carry out a master/standby changeover, or whether to conclude the operation by setting the system to redundant state.

Coupling with the objective of setting up system redundancy

In order to exclude differences in the two units, the master and the standby CPU perform the following consistency checks.

It checks the consistency of:

1. the memory configuration

2. the operating system version

3. contents of load memory on FLASH Card

4. contents of load memory in the integrated SRAM and on the RAM card)

If items 1., 2. or 3. are inconsistent, the standby CPU goes into STOP and outputs an error message.

If item 4. is inconsistent, the master CPU copies the user program from its load memory in RAM to the standby CPU.

The user program stored in load memory on the FLASH card will not be transferred.
It must be identical before you initiate coupling.

Coupling with master/standby changeover

STEP 7 supports the following options:

*   "Switch to CPU with modified configuration"

*   "Switch to CPU with expanded memory configuration"

*   "Switch to CPU with changed hardware version status"

*   "Switch to CPU via only one intact redundancy coupler"

The current operating system versions do not support the "Switch to CPU with modified operating system" option.

Switching to the CPU containing the modified configuration

You may have modified the following elements on the standby CPU:

*   the hardware configuration

*   the type of load memory (for example, you have replaced a RAM card with a FLASH card); the new load memory may be larger or smaller than the old one.

The master does not transfer any blocks to the standby during the coupling operation. Fro detailed information, refer to chapter 7.3.3.

For information on steps to be performed, based on the scenarios described earlier (modification of the hardware configuration, or of the memory medium for load memory), refer to chapter 12.

7-9

Note

Although you may not have modified the hardware configuration or the type of load memory on the standby CPU, a master/standby changeover is carried nonetheless and the previous master CPU goes into STOP.

Changing over to the CPU containing the memory expansion

Assuming you expanded load memory on the standby CPU, you need to verify that the memory media for storing load memory are identical, i.e. check that you have either used RAM Cards or FLASH Cards. The contents of FLASH cards must be identical.

During the coupling operation, the system transfers the user program blocks (OBs, FCs, FBs, DBs, SDBs) from load memory of the master to RAM on the standby CPU. Exception: if the load memory modules are located on FLASH Cards, the system only transfers the blocks from RAM..

For information on changing the type of memory module or on load memory expansion, refer to chapter 12.7.

Notice

If you have implemented a different type of load memory module or operating system on the standby CPU, this CPU will not go into RUN, but rather return to STOP and write a corresponding message to the diagnostics buffer.
Assuming you have not expanded load memory on the standby CPU, this CPU will not go into RUN, but rather return to STOP and write a corresponding message to the diagnostics buffer.
The system does not perform a master/standby changeover, and the previous master CPU will remain in RUN.

7.3.2 Update sequence

What happens during update?

The execution of communication functions and OBs is restricted by sections. Likewise, all the dynamic data (content of the data blocks, timers, counters and flags) will be transferred to the standby CPU.

Update procedure:

1. Until the update is completed, all asynchronous SFCs which access the I/O modules (SFCs 13, 51, 52, 53, 55 to 59) initiate a "negative" acknowledgment with the return values W#16#80C3 (SFCs 13, 55 to 59) or W#16#8085 (SFC 51). With these values are returned, the jobs should be repeated by the user program.

2. Message functions are delayed until the update is completed (see list below).

3. The execution of the OB1 and of all OBs up to priority class 15 is delayed.

   With watchdog interrupts, the generation of new OB requests is inhibited, i.e. no new watchdog interrupts are stored and, thus, no new request errors occur.

   The system waits until the update is completed, and then generates and process up to one request per watchdog interrupt OB. The time stamp of watchdog interrupts generated with delay can not be evaluated.

4. Transfer of all data block contents which were modified since coupling.

5. The system returns a negative acknowledgment of the following communication requests.

   – Reading/writing data records using O&M functions

   – Reading data records using SSL information

   – Disabling and enabling messages

   – Logon and logoff for messages

   – Acknowledgement of messages

   The CPU passes specific requests to other modules.

6. The system returns a negative acknowledgment of initial calls of communication functions which manipulate the contents in RAM. See also *System Software for S7-300/400, System and Standard Functions*. All remaining communication functions are executed with delay, after the update is completed.

7. The system disables the generation of new requests of all OBs of priority class > 15, i.e. new interrupts will not be saved and thus do not generate any request errors.

   Queued interrupts are not requested again and processed unless the update is completed. The time stamp of delayed interrupts can not be evaluated.

   The system no longer executes the user program or updates the I/O.

8. It generates the start event for the watchdog interrupt OB with special handling if its priority class > 15, and executes this OB as required.

Note

The watchdog interrupt OB with special handling is of particular significance in situations where you need to address certain modules or program elements within a specific time. This is a typical scenario in fail-safe systems. For details, refer to the *S7-400F and S7-400FH Programmable Controllers* and *S7-300 Programmable Controllers, Fail–safe Signal Modules* manuals.

9. Transfer of outputs and of all data block contents which were modified. Transfer of timers, counters, flags and inputs. Transfer of the diagnostics buffer content.

   During this data synchronization, the system interrupts the clock pulse for watchdog interrupts, delay interrupts and S7 timers. This results in the loss of any synchronism between watchdog and time-of-day interrupts.

10. Lift all restrictions. Delayed interrupts and communication functions will be executed. All OBs continue to be executed.

   A constant bus cycle time compared to previous calls can no longer be guaranteed for delayed watchdog interrupt OBs.

Notice

Process alarms and diagnostics interrupts are stored by the I/O. Such interrupt requests output by distributed I/O modules will be executed when the block is re–enabled. Any such requests by central I/O modules can not be executed unless the same interrupt request did not occur again as long as the status was locked.

If the PG/ES requested a master/standby changeover, the previous standby CPU assumes master mode and the previous master CPU goes into STOP when the update is completed. Both CPUs will otherwise go into RUN (redundant system status), and execute the user program in synchronism.

In the next cycle after the master/standby changeover and update, OB1 will be assigned a separate identifier. See the *System Software for S7-300/400, System and Standard Functions*) reference manual. For information on further special features derived from a modified configuration, refer to chapter 7.3.3.

Delayed message functions

The listed SFCs, SFBs and operating system services trigger the output of messages to all logged partners. Those functions will be delayed after the start of the update.

- SFC 17 "ALARM_SQ", SFC 18 "ALARM_S", SFC 107 "ALARM_DQ", SFC 108 "ALARM_D"

- SFC 52 "WR_USMSG"

- SFB 31 "NOTIFY_8P", SFB 33 "ALARM", SFB 34 "ALARM_8", SFB 35 "ALARM_8P", SFB 36 "NOTIFY", SFB 37 "AR_SEND"

- Statuses

- System diagnostics messages

From this time on, any requests to enable and disable messages by SFC 9 "EN_MSG" and SFC 10 "DIS_MSG" will be rejected with a negative return value.

Communication functions with derived requests

After it has received one of the requests specified below, the CPU must in turn generate communication requests and output these to other modules. Such derived requests may involve requests to read or write parameter data records from/to distributed I/O modules. These requests will be rejected until the update is completed.

- Reading/writing data records using O&M functions

- Reading data records by means of SSL enquiry

- Disabling and enabling messages

- Logon and logoff for messages

- Acknowledgement of messages

Note

The last three of the functions listed are registered by a WinCC system, and automatically repeated when the update operation is completed.

7.3.3 Changeover to the CPU which contains the modified configuration or memory expansion

Changeover to the CPU which contains the modified configuration

You may have modified the following elements on the standby CPU:

• the hardware configuration

• the type of memory module for load memory. You may have replaced a RAM Card with a FLASH Card. The new load memory may be larger or smaller than the old one.

For information on steps required in the scenarios mentioned above, refer to chapter 12

Note

Although you may not have modified the hardware configuration or the type of load memory on the standby CPU, a master/standby changeover is carried nonetheless and the previous master CPU goes into STOP.

When you initiate a coupling and update operation with "Switch to CPU with modified configuration" option in STEP 7, the system reacts as follows with respect to handling of the memory contents.

Load memory

It does not copy the content of load memory from the master to the standby CPU.

RAM

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

- Contents of all data blocks assigned the same interface time stamp in both load memories and the attributes "Read Only" and "unlinked".

- Data blocks generated in the master CPU by SFCs.

  The DBs generated in the standby CPU by means of SFC will be deleted.

  If a data block with the same number is found in the load memory of the standby CPU, coupling will be cancelled and a message is written to the diagnostics buffer.

- Process images, timers, counters and flags

- Diagnostics buffer

  If the configured size of the diagnostics buffer of the standby CPU is smaller than that of the master CPU, only the amount of entries configured for the standby CPU will be transferred. The most recent entries will be selected from the master CPU.

If there is insufficient memory, coupling will be cancelled and a message is written to the diagnostics buffer.

The status of SFB instances of S7 Communication contained in modified data blocks will be restored to the status prior to their initial call.

Note

When changing over to a CPU with modified configuration, the size of load memories in the master and standby may be different.

Changeover to the CPU which contains the memory expansion

Assuming you have expanded load memory in the standby CPU, you should verify that the inserted load memory module are of the same type, i.e. either RAM cards or FLASH cards. If you expanded with FLASH cards, their contents must be identical.

Notice

Assuming you have implemented a different type of load memory module or operating system on the standby CPU, this CPU will not go into RUN, but rather return to STOP and write a corresponding message to the diagnostics buffer. Assuming you have not expanded load memory on the standby CPU, this CPU will not go into RUN, but rather return to STOP and write a corresponding message to the diagnostics buffer.
The system does not perform a master/standby changeover, and the previous master CPU will remain in RUN.

For information on changing the type of memory module or on load memory expansions, refer to chapter 12.

When you initiate a coupling and update operation with "Switch to CPU with memory expansion" option in STEP 7, the system reacts as follows with respect to the handling of memory contents.

RAM and load memory

During the coupling operation, the system transfers the user program blocks (OBs, FCs, FBs, DBs, SDBs) from load memory of the master to RAM on the standby CPU. Exception: if the load memory modules are located on FLASH Cards, the system only transfers the blocks from RAM.

7.3.4 Disabling coupling and update operations

Coupling and update operations go along with a cycle time extension. Such operations include a certain time span in which the system does not update the I/O, see chapter 7.4 "Time monitoring" Make allowances for this feature in particular when using distributed I/O and n master/standby changeovers after the update operation (i.e. with modification of the configuration in run).

Caution

Always perform coupling and update operations when the process is not in a critical state.

You can set specific start times for coupling and update operations at SFC 90 "H_CTRL". For detailed information on this SFC, refer to the *System Software for S7-300/400, System and Standard Functions*) manual.

Notice

If the process generally tolerates cycle time extensions, you do not need to call SFC 90 "H_CTRL".

The CPU does not perform a self-test during coupling and update operations. In a fail–safe system, you should therefore avoid any excess delay times for the update operation. For further details, refer to the *S7-400F and S7-400FH Programmable Controllers* manual.

Example of a time–sensitive process

A slide block with a 50 mm cam moves on an axis at a constant velocity $v = 10$ km/h $= 2.78$ m/s $= 2.78$ mm/ms. A switch is located on the axis. The switch is thus actuated by the cam for the duration of $t = 18$ ms.

In order to enable the CPU to detect the actuation of the switch, the inhibit time for priority classes > 15 (see below for definition) must be clearly below 18 ms.

With respect to maximum inhibit times for operations of priority class > 15, STEP 7 only supports settings of 0 ms or 100 to 60000 ms. You thus need to work around this situation by taking one of the following measures:

• Shift the start time of coupling and update operations in order to avoid a negative impact on time–sensitive processes. Use SFC 90 "H_CTRL" to set this time (see above).

• Use a considerably longer cam and / or clearly reduce the approach velocity of the slide block.

7.4   Time monitoring

Program execution will be interrupted for a certain time when the CPU performs an upgrade. Section 7.4 is relevant to you if this period represents a critical issue to the process. If this is the case, configure one of the monitoring times described below.

During the update operation, the redundant system will monitor the cycle time extension, communication delay and the interlock time for priority classes > 15 in order to ensure that their maximum values are not exceeded, and that the configured minimum I/O retention time is maintained.

Notice

If you have not defined any default values for the monitoring times, make allowances for the update of the cycle monitoring time. If this is the situation, the update will be cancelled, and the redundant system changes over to stand–alone mode: The previous master CPU remains in RUN, and the standby CPU goes into STOP.

You can either configure all the monitoring times or none at all.

You made allowances for technological requirements in your configuration of monitoring times.

The monitoring times are described in detail below.

- Maximumcycle time extension

  - The cycle time extension is equivalent to a period of the update operation during which neither OB1,nor any further OBs up to priority class 15 are executed. The "normal" cycle time monitoring function is disabled within this time span.

  - The maximum cycle time extension represents the configured and permissible maximum.

- Maximum communication delay

  - The communication delay represents a time span within the update operation during which the CPU does not execute any communication functions. Note: the master CPU holds all online connections.

  - The maximum communication delay represents the configured and permissible maximum.

- Maximum inhibit time for priority classes > 15

  - Inhibit time for priority classes > 15: the time span within an update operation during which the CPU neither executes any OBs, nor any user programs, nor any further I/O updates.

  - The maximum inhibit time for priority classes > 15 represents the configured and permissible maximum.

- Minimum I/O retention time:

  This represents the interval between copying of the outputs from the master CPU to the standby CPU and the time of the transition to redundant system state or master/standby changeover (time at which the previous master CPU goes into STOP, and the new master CPU goes into RUN). Both CPUs control the outputs within this period, in order to prevent the I/O from going down when the system performs an update with master/standby changeover.
  The minimum I/O retention time is of particular importance when updating with master/standby changeover. If you set the minimum I/O retention time to zero, the outputs could possibly shut down when you modify the system in run.

The monitoring start times are indicated in the highlighted boxes in figure 7-2. These times expire when the systems enters the redundant state or after a master/standby changeover, i.e. with the transition of the new masters to RUN when the update is completed.

The figure below provides an overview of the relevant update times.
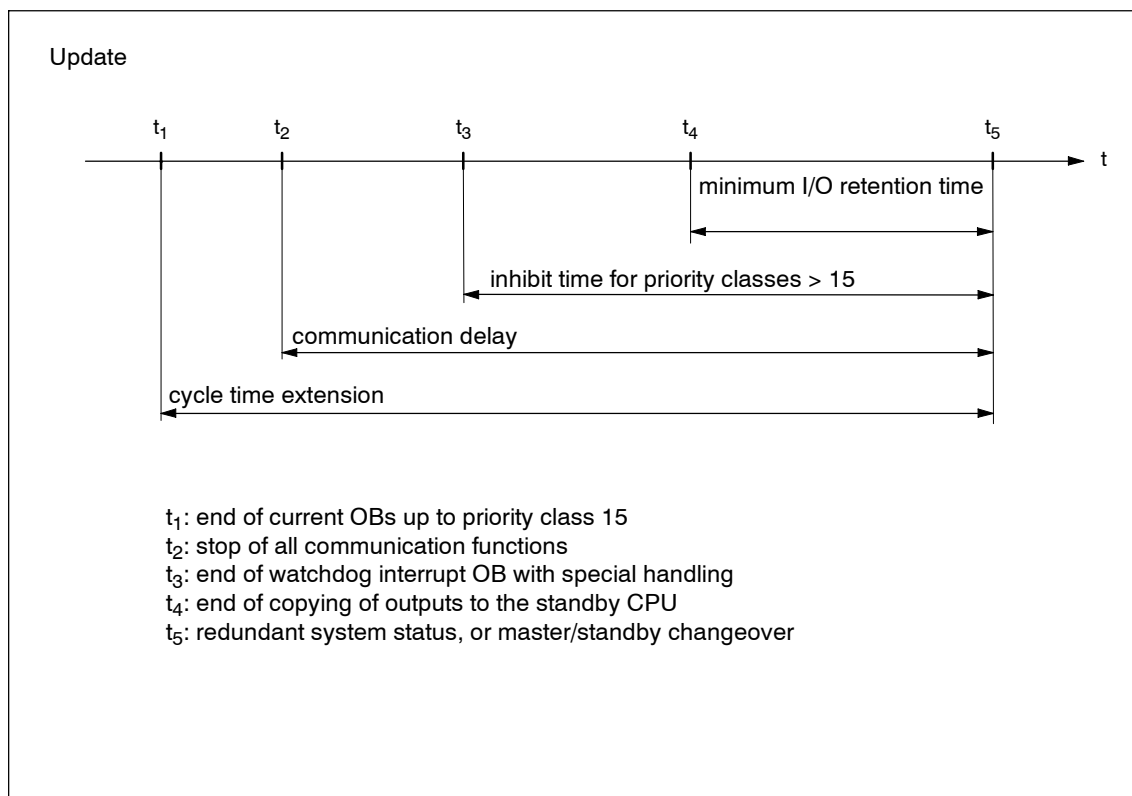
Figure 7-4   Meaning of the times relevant for updates

Reaction to timeout errors

If one of the times monitored exceeds the configured maximum, the following procedure will be initiated:

1. Cancellation of the update

2. The redundant system retains stand–alone mode, with the previous master CPU in RUN

3. The cause of cancellation will be written to the diagnostics buffer

4. Call of OB72 (with corresponding start information)

The standby CPU evaluates its system data blocks once again.
Then, and at least one minute later, the CPU retries to perform the coupling and update operation. If still unsuccessful even after 10 retries, the operation will be aborted. You must then continue by initiating a new coupling and update operation.

The monitoring timeout may have been caused by:

- high interrupt load (output by I/O modules, for example)

- high communication load causing prolonged execution times for active functions

- in the final update phase, the system has to copy large volumes of data to the standby CPU.

7.4.1   Time‑based reaction

Time‑based reaction during coupling

The influence of coupling operations on our plant control system should be kept to an absolute minimum. The current load on your automation system is therefore a decisive factor in the increase of coupling times. The time required for coupling is in particular determined by

- communication load, and the

- cycle time

The following applies to no–load operation of the automation system:

Coupling runtime = size of load memory and work memories in MB × 1 s
+ base load

The base load is formed by a few seconds.

Whenever your automation system is subject to high load, the memory-specific share may increase up to 1 minute per MB.

Time-based reaction during updates

The update transfer time is determined by the number and overall length of modified data blocks, rather than on the modified volume of data within a block. it is also determined by the current process status and communication load.

As an approximation, we can interpret the maximum inhibit time to be configured for priority classes > 15 as a function of the data volume in work memory. The volume of code in work memory is irrelevant.

7.4.2 Ascertaining the monitoring times

Determination using STEP 7 or formulas

STEP 7 version 5.2 or higher automatically calculates the monitoring times listed below for each new configuration. You may also calculate these times using the formulae and procedures described below. They are equivalent to the formulae provided in STEP 7.

- maximum cycle time extension
- maximum communication delay
- maximum retention time for priority classes
- minimum I/O retention time

You can also initiate an automatic calculation of monitoring times using the Properties CPU -> Trigger H Parameters dialog box in HW Config.

Monitoring time accuracy

Note

The monitoring times determined by STEP 7 or by using the formulae merely represent recommended values.

These times are based on a redundant system with two communication partners and an average communication load.

Your system profile may differ considerably from those scenarios. Thus, observe the following rules.

*   The cycle time extension factor may increase sharply at a high communication load.

*   Any modification of the system in run may lead to a significant increase in cycle times.

*   Any increase in the number of programs executed in priority classes > 15 (in particular those of communication blocks) will automatically increase communication delay and the cycle time extension.

*   You can even undercut the calculated monitoring times in small high–performance systems.

Configuration the monitoring times

When configuring monitoring times, always make allowances for the following dependencies; conformity will be checked by STEP 7:

max. cycle time extension
> max. communication delay
> (max. inhibit time for priority classes > 15)
> min. I/O retention time

If you have configured different monitoring times in the CPUs and perform a coupling and update operation with master/standby changeover, the system always applies the higher of the two values.

Calculating the minimum I/O retention time (T_{PH})

The following applies to the calculation of the minimum I/O retention time:

*   with central I/O: T_{PH} = 30 ms

*   with distributed I/O: T_{PH} = 3 × T_{TRmax}

    whereby T_{TRmax} = maximum target rotation time
                   at all DP master systems of the redundant station

When using central and distributed I/O, the resultant minimum I/O retention time is:

T_{PH} = MAX (30 ms, 3 × T_{TRmax})

Figure 7-5 shows the relationship between the minimum I/O retention time and the maximum inhibit time for priority classes > 15.
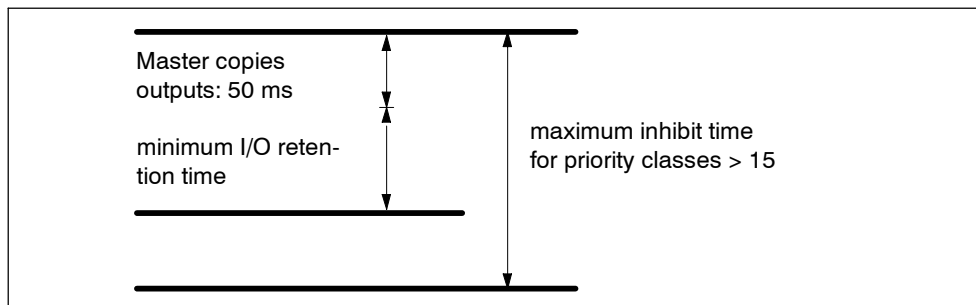
Figure 7-5   Relationship between the minimum I/O retention time and the maximum inhibit time for priority classes > 15

Note the following condition:

50 ms + minimum I/O retention time
(maximum inhibit time for priority classes > 15)

It follows that a high minimum I/O retention time may determine the maximum inhibit time for priority classes > 15.

Calculating the maximum inhibit time for priority classes > 15 (T_{P15})

The maximum inhibit time for priority classes > 15 is based on four decisive factors:

- As shown in Figure 7-2, all the contents of DBs which were modified since the last copy to the standby CPU will be transferred to the standby CPU again when the update is completed. **The number and structure of the DBs** you describe in the high–priority classes play a decisive factor in the duration of this operation, and thus in the maximum inhibit time for priority classes > 15. Relevant information is available in the remedies described below.

- In the final update phase, all OBs will either be delayed or disabled. In order to avoid any unnecessary prolongation of the maximum inhibit time for priority classes > 15 due to unfavorable programming, you should always process the particularly time–sensitive I/O components in a **selected watchdog interrupt**. This is particularly relevant in the case of fail-safe user programs. You can configure this watchdog interrupt in your project and execute it automatically immediately after the start of the maximum inhibit time for priority classes > 15, provided you have assigned it a priorityclass > 15.

- In a coupling and update operation with master/standby changeover (see chapter 7.3.1), you also need to changeover the active communication channel at the active DP slaves when the update is completed. This operation prolongs the time within which valid values can neither be read, nor be output. The duration of this operation is determined by your **hardware configuration**.

- The **technological conditions in your process** give rise to requirements with respect to the time an I/O update may be deferred. This is of particular importance to time-monitored processes in fail-safe systems.

Note

For further information on special features for applications with fail–safe modules, refer to the *S7-400F and S7-400FH Programmable Controllers* and *S7-300 Programmable Controllers, Fail–safe Signal Modules* manuals. This applies in particular to the internal runtimes of fail-safe modules.

1. In STEP 7, determine the following times derived from the bus parameters at each DP master system

    – T_{TR} for the DP master system

    –   DP changeover time (referred to below as T_{DP_UM})

2. Based on the technical data of the switched DP slaves, define the following parameters at each DP master system

    –   the maximum changeover time of the active communication channel (referred to below as T_{SLAVE_UM}).

3. Based on the technological defaults of your system, define

    –   the maximum permissible operating time of your I/O modules without update (referred to below as T_{PTO}).

4. Based on your user program, define

    –   the cycle time of the highest–priority or selected (see above) watchdog interrupt (T_{WA})

    –   the runtime of your program in this watchdog interrupt (T_{PROG})

5. The resultant status at each DP master system:

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

Note

Cancel the calculation when the value T_{P15}(DP master system) < 0. Possible remedies are shown below the following example calculation. Make the appropriate modifications, and then restart the calculation at 1.

6. Select the minimum of all T_{P15} (DP master system) values. This time is then known as T_{P15_HW}.

7. Define the share of the maximum inhibit time for I/O classes > 15 which is determined by the minimum I/O retention time (T_{P15_OD}):

$$T_{P15\_OD} = 50 \text{ ms} + \text{min. I/O retention time} \qquad [2]$$

Note

Cancel the calculation when $T_{P15\_OD} > T_{P15\_HW}$. Possible remedies are shown below the following example calculation. Make the appropriate changes, and then restart the calculation at 1.

8. Using the information inchapter 7.4.4, define the share of the maximum inhibit time for priority classes > 15 which is determined by the user program (T_{P15_AWP}).

Note

Cancel the calculation when $T_{P15\_AWP} > T_{P15\_HW}$. Possible remedies are shown below the following example calculation. Make the appropriate changes, and then restart the calculation at 1.

---

9. The recommended value for the maximum inhibit time for priority classes > 15 is now derived from:

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

Example of the calculation of T_{P15}

In the next steps,we define the maximum permitted update period during which the operating system of an existing plant configuration does not execute any programs and I/O updates.

Let there be two DP master systems: DP master system_1 is "interconnected" with the CPU via MPI/DP interface of the CPU, and DP master system_2 via an external DP master interface.

1. Based on 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. Based on the technical data of the DP slaves used:

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

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

3. Based on the technological specifications of your system:

   $T_{PTO\_1} = 1250$ ms

   $T_{PTO\_2} = 1200$ ms

4. Based on the user program:

   $T_{WA} = 300$ ms

   $T_{PROG} = 50$ ms

5. based on the formula [1]:

T_{P15} (DP master system_1)

= 1250 ms − (2 × 25 ms + 300 ms + 50 ms + 100 ms + 30 ms) = 720 ms

T_{P15} (DP master system_2)

= 1200 ms − (2 × 30 ms + 300 ms + 50 ms + 80 ms + 50 ms) = 660 ms

Check: as T_{P15} > 0, continue with

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

7. Based on the formula [2]:

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

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

8. Based on section 7.4.4 with 170 KB f user program data:

T_{P15_AWP} = 194 ms

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

9. based on formula [3], we can now derive the recommended maximum inhibit time for priority classes > 15:

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

T_{P15} = 194 ms

By setting a maximum inhibit time for priority classes > 15, i.e. 194 ms in STEP 7, you can ensure that the system detects any signal transitions during the update operation at a signal duration of 1250 ms or 1200 ms.

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

If no recommendation results from the calculation of the maximum inhibit time for priority classes > 15, you can remedy this by taking various measures:

- Reduce the watchdog interrupt cycle of the configured watchdog interrupt.

- If T_{TR} times are of a particular length, distribute the slaves to several DP master systems.

- Increase the transmission rate at the relevant DP master systems.

- Configure the DP/PA links and Y links in separate DP master systems.

- If there is a greater difference in changeover times at the DP slaves, and thus (generally) greater differences in T_{PTO}, distribute these slaves to several DP master systems.

- If you do not expect any significant interrupt of configuration load at the various DP master systems, you may also reduce the calculated T_{TR} times by approx. 20% to 30%. However, this increases the risk of a station failure at the distributed I/O.

- The time value T_{P15_AWP} represents a guideline and depends on your program structure. You may reduce it by taking the following measures, for example:

  – Save highly dynamic data to different DBs as data which not subject to frequent modification.

  – Assign the DBs a smaller length in RAM.

Whenever you reduce the T_{P15_AWP} and ignore the measures described earlier, you run the risk that the update operation will be aborted due to a monitoring timeout.

Calculation of the maximum communication delay

Its is advisable to use the following formula:

maximum communication delay =
4 x (maximum inhibit time for priority classes > 15)

Decisive factors for this time are the process status and the communication load in your system. This can be interpreted as the absolute load, or as the load in proportion to the size of your user program. You may have to adjust this time.

Calculation of the maximum cycle time extension

Its is advisable to use the following formula:

maximum cycle time extension =
10 x (maximum inhibit time for priority classes > 15)

Decisive factors for this time are the process status and the communication load in your system. This can be interpreted as the absolute load, or as the load in proportion to the size of your user program. You may have to adjust this time.

7.4.3   Influences on time‑based reactions

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 S7 communication, and of SFBs for generating block-specific messages

- system modifications in run

- settings by means of dynamic volume frameworks

- expansion of distributed I/O. A lower transmission rate and higher number of slaves increases the time required for I/O updates.

In the worst case, 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 executed until the start of interrupt delays

Explicit delay of the update

Delay the update using SFC 90 "H_CTRL", and re–enable when the system state shows less communication or interrupt load.

Caution

The update delay increases the time of stand–alone operation of the redundant system.

7.4.4 Performance values for coupling and update operations

User program share T_{P15_AWP} of the maximum inhibit time for priority classes > 15

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

T_{P15_AWP} in ms = 0.7 x <the size of DBs in RAM in KB> + 75

The table below shows the derived times for some typical values in RAM data.

Table 7-3   Typical values for the user program share T_{P15_AWP} of the max. inhibit time for priority classes > 15

RAM data	T_{P15_AWP}
500 KB	220 ms
1 MB	400 ms
2 MB	0.8 s
5 MB	1.8 s
10 MB	3.6 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.
  In particular for fail–safe systems, this calculated value must be more precise in order to avoid any timeout of driver blocks (see chapter 7.4.2).

- For active or queued communication requests, allowances are made for an update time of approximately 100 ms per MB of data in the RAM area used by DBs.
  Depending on the communication load of your automation system, you have to add to or deduct a certain value when you set T_{P15_AWP}.

7.5 Special features in coupling and update operations

Requirement of input signals during the update

Any process signals read previously are retained and not included in the update. During the update operation, the CPU will only recognize changes of process signals if their status remains static until the update is completed.

The CPU does not detect pulse (signal transitions "0 → 1 → 0" or "1 → 0 → 1") signals which are generated during the update.

You should therefore ensure that the interval between signal transitions (pulse period) is always greater than the required update period.

Communication links and functions

Connections of the master CPU will not be shut down. However, the CPU does not execute any associated communication requests until the update is completed, and rather queue these for execution when the following states are reached:

* the update is completed, and the system is in redundant state

* the update and master/standby changeover are completed, the system is in stand–alone mode

* the update was aborted (due to timeout, for example), and the system has returned to stand–alone mode.

An initial call of communication blocks is not possible during the update.

CPU request of memory reset due to an aborted coupling operation

The standby CPU will always request a memory reset when the coupling operation is aborted while the master copies the content of load memory to the standby CPU. This indicted in the diagnostics buffer by event ID W#16#6523.

Using I/O on the S7-400H

<div align="right"># **8**</div>

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.

8.1 Introduction

I/O configuration types

In addition to the power supply module and CPUs, which are always redundant, the operating system supports the following I/O configurations:

I/O type	Configuration	Availability
Digital input	Single-channel one-sided Single-channel switched Dual-channel redundant	normal increased high
Digital output	Single-channel one-sided Single-channel switched Dual-channel redundant	normal increased high
Analog input	Single-channel one-sided Single-channel switched Dual-channel redundant	normal increased high
Analog output	Single-channel one-sided Single-channel switched Dual-channel redundant	normal increased high

A dual-channel redundant configuration on user level is also possible. However, you have to implement redundancy in the user program (refer to Section 8.5).

Addressing

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

Limits of I/O configuration

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

Module racks with even number are always assigned to CPU 0, and racks with odd numbers are assigned to CPU 1.

For applications with distributed I/O, each one of the CPU units supports the connection of up to 12 DP master systems, i.e. two DP master systems on the integrated interfaces of the CPU, plus 10 via external DP master systems.

The integrated MPI/DP interface supports the operation of up to 32 slaves. You can connect up to 125 distributed I/O devices to the integrated DP master interface and to the external DP master systems.

8.2   Using single-channel, one-sided I/O

Definition of single-channel one-sided I/O

A single-channel, one-sided configuration contains only one set of I/O modules (single-channel) which is installed only in one of the two units, and only addressed by this unit.

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

- central racks and expansion devices
- distributed I/O devices

An installation with single-channel, one-sided I/O is useful for the operation of stand-alone I/O channels and system components which only require the standard availability.
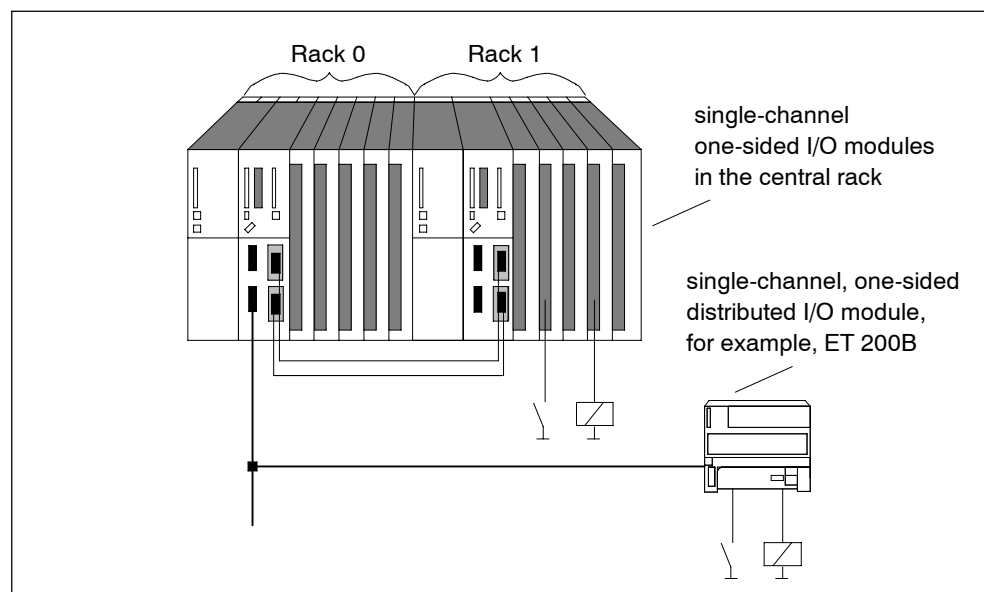
Figure 8-1   Single-channel, one-sided I/O configuration

Single-channel, one-sided I/O and the user program

When the system is in redundant state, the data read from one-sided components such as digital inputs are transferred automatically to the second system unit.

When the transfer is completed, the data read from the single-channel one-sided I/O are available on both units, and can be evaluated in their identical user programs. For data processing in redundant system state, it is thus irrelevant whether the I/O is connected to the master or to the standby CPU.

In stand-alone mode, access to the one-sided I/O assigned to the partner unit is not possible. Make allowances for this factor in your program by assigning functions to the single-channel one-sided I/O which can only be executed conditionally. This ensures that specific I/O access functions are only called in redundant system state, and in stand-alone mode at the relevant unit.

Notice

The user program also has to update the process image for single-channel, one-sided output modules when the system is in stand-alone mode (direct access, for example). Any process image partitions used must be updated accordingly (SFC27 "UPDAT_PO") by the user program in OB72 (recovery of redundancy). The system would otherwise initialize the single-channel one-sided output modules of the standby CPU with the old values after the system transition to redundant state.

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

The redundant system with single-channel, one-sided I/O reacts to errors as a standard S7-400 system, i.e.:

- the I/O is no longer available after its failure.

- the entire process I/O of a system unit is no longer available if this unit fails.

8.3 Using single-channel switched I/O

Definition of single-channel switched I/O

A single-channel switched configuration contains only one set of I/O modules (single-channel).

In redundant mode, these can addressed by both system units.

In stand-alone mode, the master unit can always address **all switched I/O** (in contrast to one-sided I/O).

The system supports single-channel switched I/O configurations containing an ET 200M distributed I/O module with active backplane bus and a redundant PROFIBUS DP slave interface module.

You can use the following interfaces:

Table 8-1   Interfaces for the deployment of single-channel peripherals

Interface	Order number
IM 153-2	6ES7 153-2BA81-0XB0
	6ES7 153-2BA01-0XB0
	6ES7 153-2BA00-0XB0
IM 153-2FO	6ES7 153-2AB02-0XB0
	6ES7 153-2AB01-0XB0
	6ES7 153-2AB00-0XB0
	6ES7 153-2AA02-0XB0

Each subsystem of the S7-400H is connected (via a DP master interface) with one of the two DP slave interfaces of the ET 200M.

PROFIBUS PA can be interconnected with a redundant system using DP/PA-Link.

You can use the following DP/PA links:

DP/PA link	Order number
IM 157	6ES7 157-0AA82-0XA0
	6ES7 157-0AA81-0XA0
	6ES7 157-0AA80-0XA0
ET 200 M as DP/PA link with	6ES7 153-2BA81-0XB0

A single-channel DP master system can be interconnected with a redundant system y means of Y-coupler. Supported Y-coupler: 6ES7 197-1LB00 0XA0

The single-channel switched I/O configuration is recommended for system components which tolerate the failure of individual modules within the ET 200M.
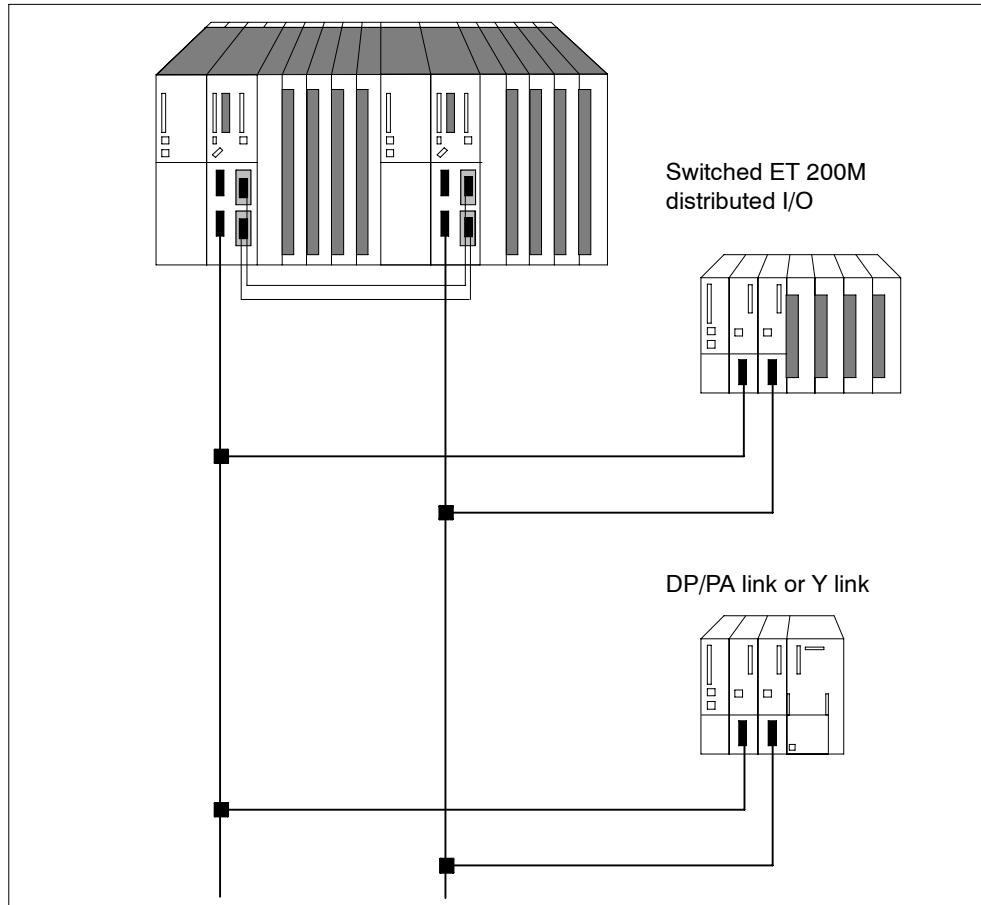
Figure 8-2　　Single-channel, switched ET 200M distributed I/O

Rule

A single-channel switched I/O configuration must always be symmetrical, i.e.:

- The redundant CPU and further DP masters must be installed in corresponding slots in both systems units (for example, slot 4 on both system units), or

- the DP masters must be connected to the same integrated interface on both system units (or example, to the PROFIBUS DP interfaces of the two redundant CPUs).

Single-channel switched I/O and the user program

In redundant mode, each subsystem can basically access any single-channel switched I/O. The data are automatically transferred via synchronization link and compared. An identical value is available to the two system units at all times owing to the synchronized access.

The S7-400H uses only 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 is always active on 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

Reaction of the S7-400H with single-channel switched I/O to errors:

- the I/O is no longer available after its failure.

- In certain failure situations, such as the failure of a system unit, DP master system or DP slave interface IM153-2 or IM 157, see chapter 9) the single-channel switched I/O remains available to the process.
  This is achieved by means of a changeover between the active and passive channel. This changeover takes place separately at each DP station. With respect to failures, we distinguish between

  – failures affecting only one station, such as the failure of the DP slave interface of the currently active channel

  – failures affecting all the stations in a DP master system.
  This includes the disconnection of the DP master interface, shutdown of the DP master system (for example, during RUN-STOP transition on a CP 443-5) and short-circuits on the cable segment 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 assumes the active state. A redundancy loss is reported to the user program by the call of OB 70 (event W#16#73A3).

The system recovers the redundant state after the fault has been remedied. This again initiates a call of OB 70 (event W#16#72A3). In this situation, a changeover between the active and slave channel does not take place.

If one channel has already failed, and the remaining (active) channel also fails, then there is a complete station failure. This initiates a call of OB 86 (event W#16#39A3).

Note

If the DP master interface module detects a failure of the entire DP master system (due to short-circuit, for example), it reports only this event ("Master system failure coming" W#16#39C3), and the operating system no longer reports individual station failures. This feature can be sued to accelerate the changeover between the active and passive channel.

Duration of a changeover of the active channel

The maximum changeover time is

DP error detection time + DP changeover time + changeover time of the DP slave interface

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

Notice

When using fail-safe modules, always set a monitoring time for each fail-safe module which is longer than the changeover time of the active channel in the redundant system. If you ignore this rule, you risk failure of the fail-safe modules during the changeover of the active channel.

Notice

The above calculation also includes the processing time in OB 70 or OB 86. Make sure that the processing time for a DP station **does not take any longer than 1 ms**. In situations requiring extensive processes, isolate these processes from direct execution of the OBs mentioned.

Note that the CPU can only detect a signal transition if the signal duration is greater than the set changeover time.
When a changeover of the entire DP master system takes place, the changeover time of the slowest component applies to all DP components. A DP/PA-Link or Y-Link usually determines the changeover time and the associated minimum signal duration. It is thus advisable to connect the DP/PA and Y links to a separate DP master system.

When using fail-safe modules, always set a monitoring time for each fail-safe module which is longer than the changeover time of the active channel in the redundant system. If you ignore this rule, you risk failure of the fail-safe modules during the changeover of the active channel.

Changeover of the active channel during a coupling and updateoperation

During a coupling and updateoperation with master/standby changeover (see section 7.3.1), the active and slave channels are changed over at all the stations of the switched I/O. OB 72 is called in this process.

Bumpless changeover of the active channel

In order to prevent failure of the I/O or the output of substitution values by the I/O during the changeover between the active and passive channel, the DP stations of the switched I/O will set their outputs to HOLD until the changeover is completed and the new channel has taken control f the process.

In order to ensure the detection of total failures of a DP station during the changeover operation, the changeover is monitored by the various DP stations and the DP master system.

Provided the minimum I/O retention time is set correctly (see chapter 7.4), any interrupts or loss of data records during the changeover are not to be expected. Automatic repetition takes place if necessary.

System configuration and programming

You should allocate switched I/O with different transfer times to separate segments in order to simplify the calculation of monitoring times, for example.

8.4 Connecting redundant I/O

Definition of redundant I/O

I/O modules are considered redundant when the system contains two sets of each module, and these are configured and operated as redundant pairs. The use of redundant I/O provides the highest degree of redundancy, because the system tolerates the failure of a CPU or of a signal module.

Configurations

The following redundant I/O configurations are supported:

1. Redundant signal modules in the central and expansion racks
   The signal modules are installed in pairs in the CPU 0 and CPU 1 units.
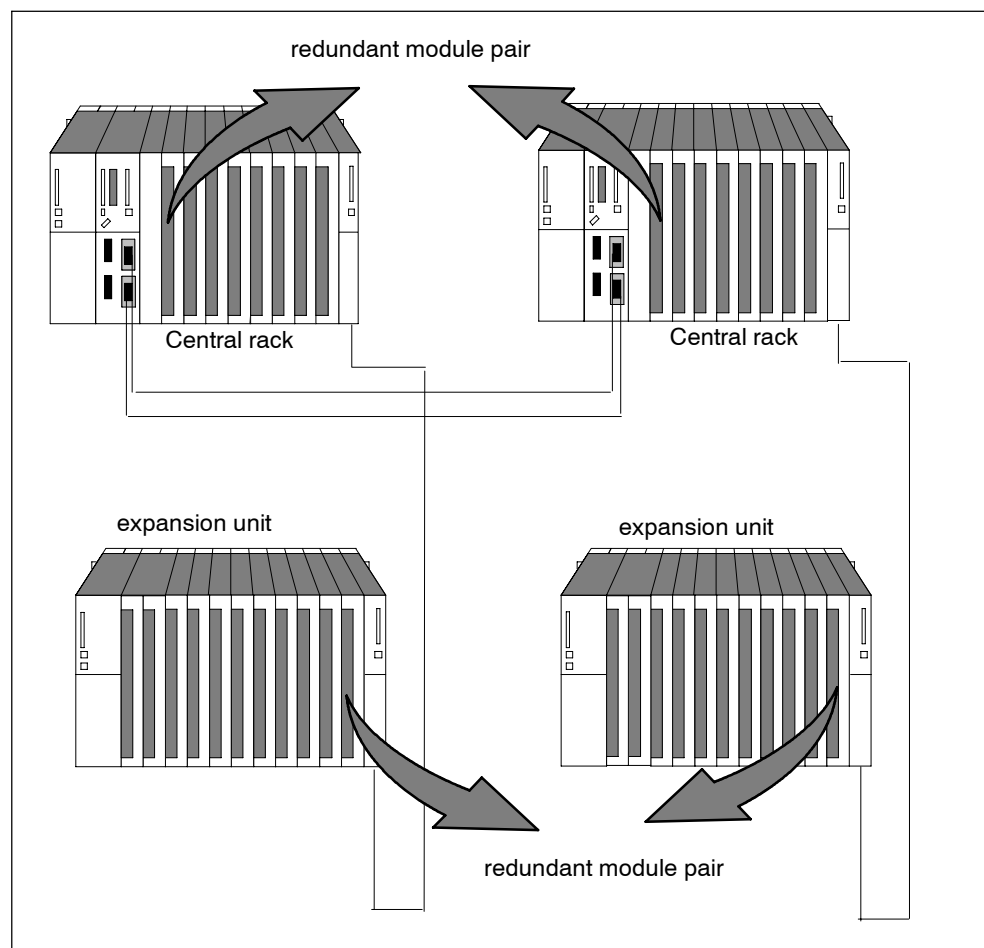
Figure 8-3   Redundant I/O in the central and expansion racks

2. Redundant I/O in the one-sided DP slave

   In distributed I/O devices ET 200M with active backplane bus the signal modules are installed in pairs.
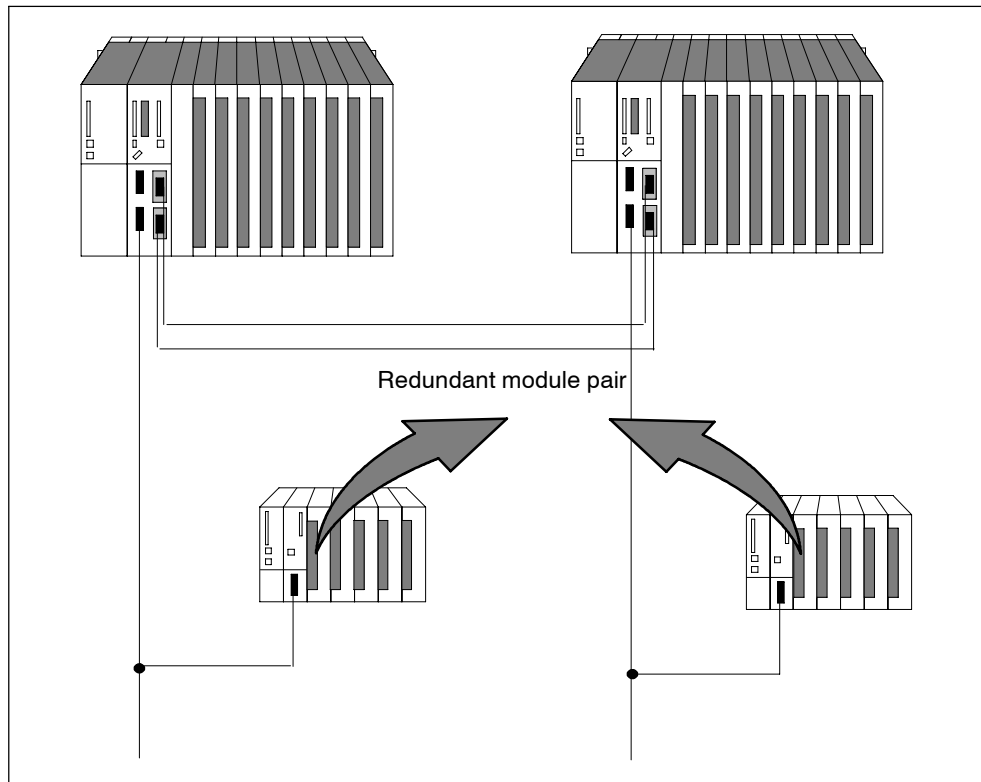
Figure 8-4   Redundant I/O in the one-sided DP slave

3. Redundant I/O in the switched DP slave

In distributed I/O devices ET 200M with active backplane bus the signal modules are installed in pairs.
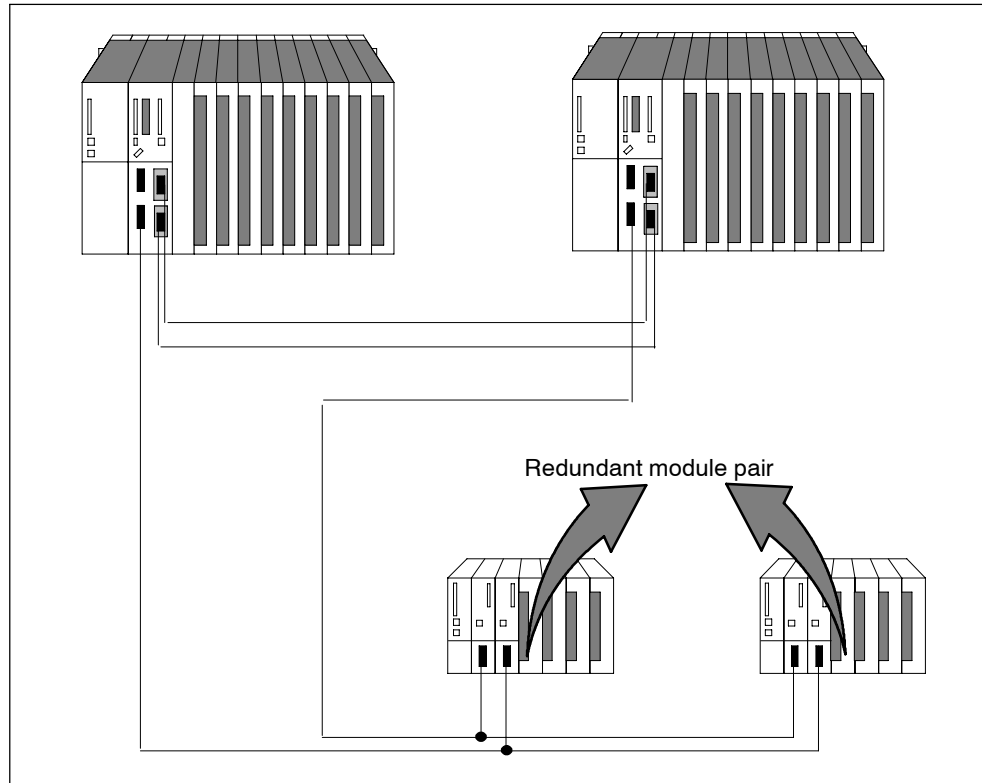
Figure 8-5   Redundant I/O in the switched DP slave

4. Redundant I/O connected to a redundant CPU operating in stand-alone mode
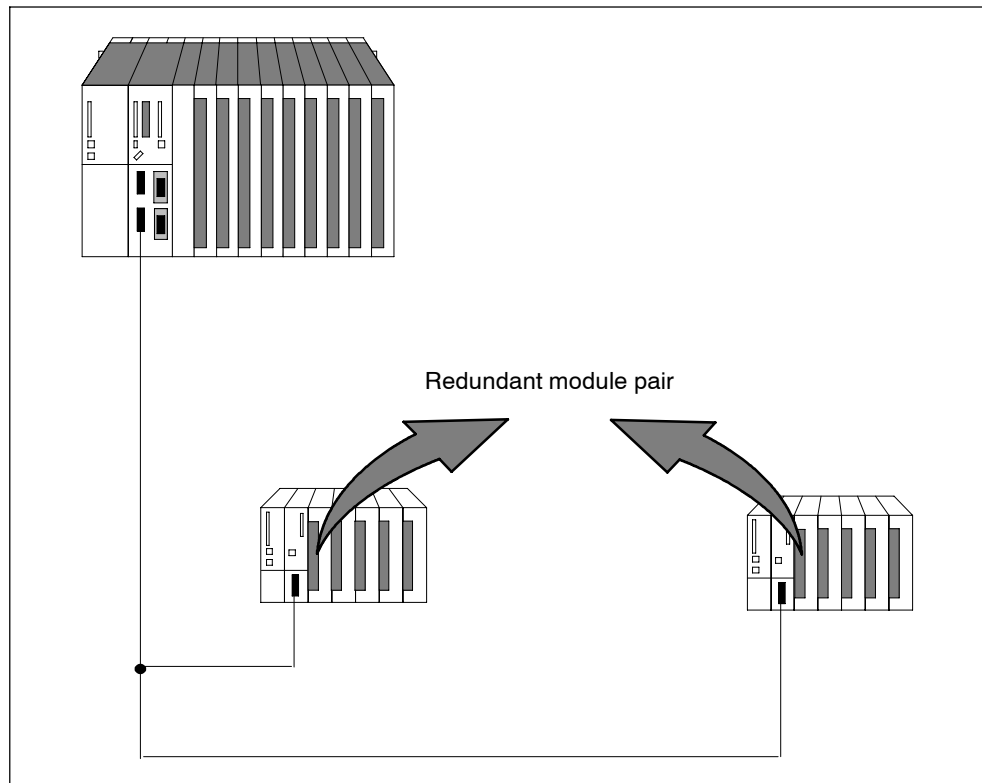
Figure 8-6   Redundant I/O in stand-alone mode

Module-granular redundancy and channel-granular redundancy

You can specify whether you operate redundant assemblies with module-granular redundancy or with channel-granular redundancy. There are two module libraries "Functional peripheral redundancy" for this.

Principle of the module-granular redundancy

Redundancy always applies to the entire module, rather than to individual channels. When a channel error occurs on the first redundant module, the entire module and its channels change over to passive state. When a channel error occurs at the second module before the initial error was eliminated and the module depassivated, this second error can not be handled by the system.

Principle of the channel-granular redundancy

Channel errors, whether due to discrepancy or diagnosis alarm (OB82), do not lead to deactivation of the entire module. Instead, only the channel concerned is deactivated. Deactivation deactivates the channel concerned as well as the assemblies deactivated due to module faults. Channel-granular deactivation significantly increases the availability for the following cases:

- Relatively frequent sensor failures

- Repairs that take a long time

- A number of channel errors on one module

Module libraries "Functional I/O redundancy"

The module libraries "Functional I/O redundancy" that support the redundant peripherals each contain the following modules:

- FC 450 "RED_INIT": Initialization function

- FC 451 "RED_DEPA": Initiate depassivation

- FB 450 "RED_IN": Function block for reading redundant inputs

- FB 451 "RED_OUT": Function block for controlling redundant outputs

- FB 452 "RED_DIAG": Function block for diagnostics of redundant I/O

- FB 453 "RED_STATUS": Function block for redundancy status information

Configure the numbers of the management data blocks for the redundant I/O in HW Config properties CPU –> H Parameter. Assign free DB numbers to these data blocks. The data blocks are created by the FC 450 "RED_INIT" during CPU startup. The default setting for the numbers of the management data blocks is 1 and 2. These data blocks do not deal with the instance data blocks of FB 450 "RED_IN" or FB 451 "RED_OUT".

The modules you use for module-granular redundancy are located in the library "Redundant IO (V1)" under STEP 7\S7_LIBS\RED_IO.

The modules you use for channel-granular redundancy are located in the library "Redundant IO CGP" under STEP 7\S7_LIBS\RED_IO, SIMATIC Manager –> "File –> Open –> Libraries".

The functions and deployment of the modules are described in the corresponding online help.

Notice

Only use modules from one or the other library. The simultaneous use of modules from both libraries is not permitted.

Switch from module-granular redundancy to channel-granular redundancy

To activate the channel-granular deactivation, you have to stop the automation system (original delete and reload user program in STOP).

Observe the following points:

It is not permitted to mix modules from the libraries "Redundant IO (V1)" and "Redundant IO CGP" in one CPU and this can lead to unpredictable characteristics.

Expand the existing projects either with modules from the library "Redundant IO (V1)" or switch completely to the library "Redundant IO CGP".

The FB452 RED_DIAG now calls up the SFB 54. For this reason, error information has been supplemented for this FB. On switching, check your calling modules.

On changing a project, ensure that all modules with the designations FB450-453 and FC450-451 have been deleted from the module folder. Perform this step in every relevant program. Translate and load your project.

Using the blocks

Before you use the blocks, configure the redundant modules in HW Config by setting these for operation in redundant mode.

Install the blocks from the "Redundant IO" library in the OBs which address the redundant modules.

In which OBs you need to install the various blocks are listed in the table below:

Block	OB
FC 450 RED_INIT	• OB 72 "CPU redundancy error" FC 450 is only executed after start event B#16#33:"Standby-master changeover by operator" • OB 80 "Timeout error" FC 450 is only executed after start event B#16#0A: "Resume RUN after reconfiguring". • OB 100 "warm restart" • OB 102 "cold restart" Call FC 450 in OB 80 if you connect redundant I/O to a redundant CPU operating in stand-alone mode.
FC 451 "RED DEPA"	When you call FC 451 in OB 83 after you inserted any modules, this function allows automatic depassivation after repairs (optional).
FB 450 "RED_IN"	• OB1 "cyclic program" • OB 30 to OB 38 "watchdog interrupt"
FB 451 "RED_OUT"	• OB1 "cyclic program" • OB 30 to OB 38 "watchdog interrupt"

Block	OB
FB 452 "RED_DIAG"	• OB 72 "CPU redundancy error" • OB 82 "diagnostics interrupt" • OB 83 "removal/insertion interrupt" • OB 85 "program runtime error" Call FC 452 in OB 83 if you connect redundant I/O to a redundant CPU operating in stand-alone mode.
FB 453 "RED_STATUS"	

In order to be able to address redundant modules by means of process image partition in watchdog interrupts, the relevant process image partition must be assigned to this pair of modules and to the watchdog interrupt. Call FB 450 "RED_IN" in this watchdog interrupt before you call the user program. Call FB 451 "RED_OUT" in this watchdog interrupt after you called the user program.

Note

Deployment of the FB 450 "RED_IN" and 451 "RED_OUT"

You have to use a separate subprocess image for each priority class used (OB1, OB 30 ... OB 38).

Hardware installation and configuration of the redundant I/O

To use redundant I/O:

1. Insert all modules required for redundancy. Pay attention to the following default rules for the configuration.

2. Configure the module redundancy using HW Config in the object properties of the relevant module.

3. Either look for a partner module for each module, or accept the default settings

   In a centralized configuration: insert the module in slot X of the rack with even number, and the redundant module at the same slot position in the next rack with odd number.
   If the module in the odd rack is inserted in slot X, the same slot in the preceding even rack is suggested for the module.

   Distributed installation in a one-sided DP slave: If the module is inserted in slot X of the slave and the DP master system is in redundant state, you should also install the module in slot X of the slave with the same PROFIBUS address in the partner DP subsystem.

   Distributed installation in a switched DP slave, stand-alone mode: If the module in the slave is inserted with a DP address in slot X, the module in the slave with the next PROFIBUS address at slot X is suggested.

4. Enter the remaining redundancy parameters for the input modules.

Notice

Always switch off power to the station or rack before you remove a redundant digital input module which is not passivated and does not support diagnostics functions. Youmight otherwise passivate the wrong module. The replacement of the front connector of a redundant module is a good example of the necessary procedure.

The valid value that can be processed by the user program are always located at the lower address of both redundant modules. This way only the lower address can be used by the application; the values of the higher address are not relevant for the application.

Notice

Redundant modules must be set up in the process image of inputs or outputs, and are always accessed using the relevant process image.

Configure the DBs for the redundant I/O, and assign the DBs free DB numbers. These DBs do not represent instance DBs of FB 450 "RED_IN" or FB 451 "RED_OUT".

Notice

When using redundant modules, select the "Cycle/Flag" tab from "HW Config -> Properties CPU 41x-H" and set up the following parameters:

"OB 85 call on I/O access error -> Only incoming and outgoing errors"

Signal modules for redundancy

The signal modules listed below can be used in channel-granular redundancy.

Table 8-2   Signal modules for channel-granular redundancy

Modules	Order number
DI16xDC 24 V	6ES7 321-7BH01-0AB0
As of product status 2, this module can also be deactivated on a granular channel group basis. In the event of an error on one channel, the entire group (2 channels) is deactivated.	
DO 16xDC 24 V/0,5 A	6ES7322-8BH01-0AB0
Tthis module can also be operated in channel-granular redundancy.	
AI 8x16Bit	6ES7 331-7NF00-0AB0
As of product status 10, this module can also be operated in channel-granular redundancy.	
AO8x12 Bit	6ES7 332-5HF00-0AB0
As of product status 5, this module can also be operated in channel-granular redundancy.	

The signal assemblies listed below can be used as redundant I/O. Observe the current notes on deployment of the assemblies in the readme file and in the SIMATIC FAQs at `http://www.siemens.com/automation/service&support` under the key word "Redundant I/O".

Table 8-3   Signal modules for redundancy

Modules	Order number
Central: dual-channel redundant DI	
DI 16xDC 24V alarm	6ES7 421-7BH01-0AB0
Use with non-redundant encoder • This module supports the "wire break" diagnostics function. To be able to use this function, make sure that when using one and two inputs these are loaded with an accumulated quiescent current in signal status "0" between 2.4 mA and 4.9 mA. You achieve this by installing a resistive load at the encoder. The value depends on the type of switch, and usually ranges between 6800 and 8200 Ohm for contacts. For Beros, calculate the resistor based on this formula: $(30V / (4.9\ mA - I\_R\_Bero)) < R < (20V / (2.4\ mA - I\_R\_Bero)$	
DI 32xDC 24 V	6ES7 421-1BL0x-0AA0
DI 32xUC 120V	6ES7 421-1EL00-0AA0

Table 8-3   Signal modules for redundancy, continued

Modules	Order number
Distributed: Redundant DI dual-channel	
DI16xDC 24 V, interrupt	6ES7 321-7BH00-0AB0
DI16xDC 24 V	6ES7 321-7BH01-0AB0
Use with non-redundant encoder • This module supports the "wire break" diagnostics function. To be able to use this function, make sure that when using one and two inputs these are loaded with an accumulated quiescent current in signal status "0" between 2.4 mA and 4.9 mA. You achieve this by installing a resistive load at the encoder. The value depends on the type of switch, and usually ranges between 6800 and 8200 Ohm for contacts. For Beros, calculate the resistor based on this formula: $(30V \, / \, (4.9 \text{ mA} - I\_R\_Bero) \; < \; R < \; (20V \, / \, (2.4 \text{ mA} - I\_R\_Bero)$	
DI16xDC 24 V	6ES7 321-1BH02-0AA0
DI32xDC 24 V	6ES7 321-1BL00-0AA0
DI 8xAC 120/230V	6ES7 321-1FF01-0AA0
DI 4xNAMUR [EEx ib]	6ES7 321-7RD00-0AB0
You cannot use the module for Ex applications in redundant mode. Use with non-redundant encoder • You may only connect 2-wire NAMUR encoders or contact encoders. • The encoder circuit should always be grounded at a common point, preferably encoder minus. • Compare the properties of the selected encoders with the specified input characteristics. This function always has to be ensured, regardless whether you are using on or two inputs. Example of valid values for NAMUR encoders: for "0" current > 0.2 mA, and for "1" current > 4.2 mA.	
DI 16xNamur	6ES7 321-7TH00-0AB0
Use with non-redundant encoder • The encoder circuit should always be grounded at a common point, preferably encoder minus. • Always operate the redundant modules on a common load power supply. • Compare the properties of the selected encoders with the specified input characteristics. This function always has to be ensured, regardless whether you are using on or two inputs. Example of valid values for NAMUR encoders: for "0" current > 0.7 mA, and for "1" current > 4.2 mA.	
DI 24xDC 24 V	6ES7 326-1BK00-0AB0
F-module in standard operation	
DI 8xNAMUR [EEx ib]	6ES7 326-1RF00-0AB0
F-module in standard operation	
Central: dual-channel redundant DI	
DO 32xDC 24V/0.5A	6ES7 422-7BL00-0AB0
A definite evaluation of the diagnostics information "P short-circuit" and "M short-circuit" is not possible.	
DO 16xAC 120/230V/2A	6ES7 422-1FH00-0AA0

Table 8-3   Signal modules for redundancy, continued

Modules	Order number
Distributed: Redundant DO dual-channel	
DO8xDC 24 V/0.5 A	6ES7 322-8BF00-0AB0
A definite evaluation of the diagnostics information "P short-circuit", "M short-circuit" and "wire break" is not possible. Deselect these separately in your configuration.	
DO8xDC 24 V/2 A	6ES7 322-1BF01-0AA0
DO32xDC 24 V/0.5 A	6ES7 322-1BL00-0AA0
DO8xAC 120/230 V/2 A	6ES7 322-1FF01-0AA0
DO 16x24 V/10 mA [EEx ib]	6ES7 322-5SD00-0AB0
You cannot use the module for Ex applications in redundant mode.	
DO 16xDC 24V/0.5A	6ES7 322-8BH01-0AB0
• The load circuit should always be grounded at a common point, preferably load minus. • A channel diagnosis is not possible.	
DO 10xDC 24 V/2 A as of product status 3	6ES7326-2BF01-0AB0
Each of the inputs and outputs must have the same address.	
Central: dual-channel redundant AI	
AI 6x16 Bit	6ES7 431-7QH00-0AB0
Use with voltage measurement • Always disable the "wire break" diagnostics function in HW Config when operating the modules with measuring transducers or thermocouples. Use with indirect current measurement • Use a 250 Ohm resistance to map the current to a voltage. See page 8-30 Use with direct current measurement • Suitable Zener diode BZX85C6v2 or 1N4734A (6.2 V because of the 50 Ohm input resistance) • Load capability of 4-wire measuring transducers: R_L > 325 Ohm (worst-case: 1 input + 1 Zener diode at an S7 overshoot value 24 mA to $R_L = (R_I * I_{max} + V_{z\,max}) / I_{max}$) • Input voltage of 2-wire measuring transducers: $V_{in\text{-}2Dr}$ < 8 V (worst-case: 1 input + 1 Zener diode at an S7 overshoot value 24 mA to $V_{in\text{-}2Dr} = R_E * I_{max} + V_{z\,max}$) **Note:** The circuit shown in Fig. 8-10 works only with active (4-wire) measuring transducers, or with passive (2-wire) measuring transducers with external power supply. Always configure the module channels for operation as "4-wire measuring transducer", and set the measuring range cube to "C" position. The module (2DMU) does not supply power to the measuring transducers.	

Table 8-3 Signal modules for redundancy, continued

Modules	Order number
Distributed: Redundant AI dual-channel	
AI8x12Bit	6ES7 331-7KF02-0AB0
Use with indirect current measurement • The rated accumulated input resistance of 100 kOhm can be reduced to 50 kOhm in the measuring ranges > 2.5 V by operating two inputs in a parallel circuit. • Always disable the "wire break" diagnostics function in HW Config when operating the modules with measuring transducers or thermocouples. • Use a 50 Ohm or 250 Ohm resistance to map the current to a voltage. See page 8-29 • This module is not suitable for direct current measurements Use of redundant encoders: • You can use a redundant sensor for the with the following voltage settings: +/- 80 mV　(only without wire break monitoring) +/- 250 mV　(only without wire break monitoring) +/- 500 mV　(wire break monitoring not configurable) +/- 1 V　(wire break monitoring not configurable) +/- 2.5 V　(wire break monitoring not configurable) +/- 5 V　(wire break monitoring not configurable) +/- 10 V　(wire break monitoring not configurable 1...5 V　(wire break monitoring not configurable)	
AI 8x16Bit	6ES7 331-7NF00-0AB0
Use with voltage measurement • Always disable the "wire break" diagnostics function in HW Config when operating the modules with measuring transducers or thermocouples. Use with indirect current measurement • Use a 250 Ohm resistance to map the current to a voltage. See page 8-30 Use with direct current measurement • Suitable Zener diode BZX85C8v2 or BZX85C8v2 (8.2 V because of the 250 Ohm input resistance) • Circuit-specific additional error: if one module fails, the other may suddenly show an additional error of approx. 0.1% • Load capability of 4-wire measuring transducers: R_L > 610 Ohm (worst-case: 1 input + 1 Zener diode at an S7 overshoot value 24 mA to $R_L = (R_I * I_{max} + V_{z\,max}) / I_{max})$ • Input voltage of 2-wire measuring transducers: $V_{in\text{-}2Dr}$ < 15 V (worst-case: 1 input + 1 Zener diode at an S7 overshoot value 24 mA to $V_{in\text{-}2Dr} = R_E * I_{max} + V_{z\,max})$	

Table 8-3   Signal modules for redundancy, continued

Modules	Order number
AI 4x15Bit [EEx ib]	6ES7 331-7RD00-0AB0
You cannot use the module for Ex applications in redundant mode. This module is suitable for voltage measurements only with redundant encoders. It is not suitable for indirect current measurements Use with direct current measurement • Suitable Zener diode BZX85C6v2 or 1N4734A (6.2 V because of the 50 Ohm input resistance) • Circuit-specific additional error: -- • Load capability of 4-wire measuring transducers: RL > 325 Ohm worst-case: 1 input + 1 Zener diode at an S7 overshoot value 24 mA RL = (RI * I_{max} + $V_{z\ max}$) / I_{max} • Input voltage of 2-wire measuring transducers: Vin-2Dr < 8 V worst-case: 1 input + 1 Zener diode at an S7 overshoot value 24 mA to Vin-2Dr = RI * I_{max} + $V_{z\ max}$ **Note:** the module supports only 4-wire measuring transducers and 2-wire measuring transducers with external power supply. The internal power supply for measuring transducers can not be used in the circuit shown in Fig. 8-10, because this outputs only 13 V, and in the worst case would thus supply only 5 V to the measuring transducer	
AI 6x13Bit	6ES7 336-1HE00-0AB0
F-module in standard operation	
Distributed: Redundant AO dual-channel	
AO4x12 Bit	6ES7 332-5HD01-0AB0
AO8x12 Bit	6ES7 332-5HF00-0AB0
AO4x0/4...20 mA [EEx ib]	6ES7 332-5RD00-0AB0
You cannot use the module for Ex applications in redundant mode.	

Notice

You need to install the F Configuration Pack for F-modules. The F Configuration Pack can be downloaded free of charge from the Internet.
You can find it at Customer Support under
`http://www.siemens.com/automation/service&support`

Which errors can be handled using redundant I/O?

There are three quality levels for the reliable operation of a redundant configuration of signal modules:

• Highest quality with fail-safe signal modules (but without F functionality)

• Medium quality with signal modules capable of diagnostics

• Simple quality with signal modules without diagnostics

Using digital input modules as redundant I/O

The following parameters are set to configure digital input modules for redundant operation:

- Discrepancy time (maximum allowed time in which the redundant input signals may differ)
  When there is still a discrepancy in the input values after the configured discrepancy time has expired, a fault has occurred.

- Reaction of the H system to discrepancy in the input values

First, the input signals of the paired redundant modules are checked to see if they match. If the values match, the uniform value is written to the lower data memory area of the process image of inputs. If there is a discrepancy and it is the first discrepancy, it is marked and the discrepancy time is started.

During the discrepancy time, the most recent matching (non-discrepant) value is written to the process image of the module with the lower address. This procedure is repeated until the values once again match within the discrepancy time or until the discrepancy time of a bit has expired.

If the discrepancy continues past the expiration of the configured discrepancy time, a fault has occurred.

The localization of the defective page is performed according to the following strategy:

1. During the discrepancy time the most recent matching value is retained as a result.

2. Once the discrepancy time has expired the following error message is displayed:
   Error code 7960: "Redundant I/O: discrepancy time at digital input expired, error not yet localized". Passivation is not performed and no entry is made in the static error image. Until the next signal transition occurs, the configured reaction is performed after the discrepancy time expires.

3. If another signal transition occurs, the module/channel in which the transition takes place is the intact module/channel and the other module/channel is passivated.

Notice

The time that the system actually needs to determine a discrepancy depends on several factors: bus transit times, cycle and call times in the user program, conversion times etc. Redundant input signals may therefore be longer than the difference in the configured discrepancy time.

Modules with diagnostics functions are passivated with the call of OB82.

Using redundant digital input modules with non-redundant sensors

You install digital input modules with non-redundant sensors in a 1-out-of-2 configuration:
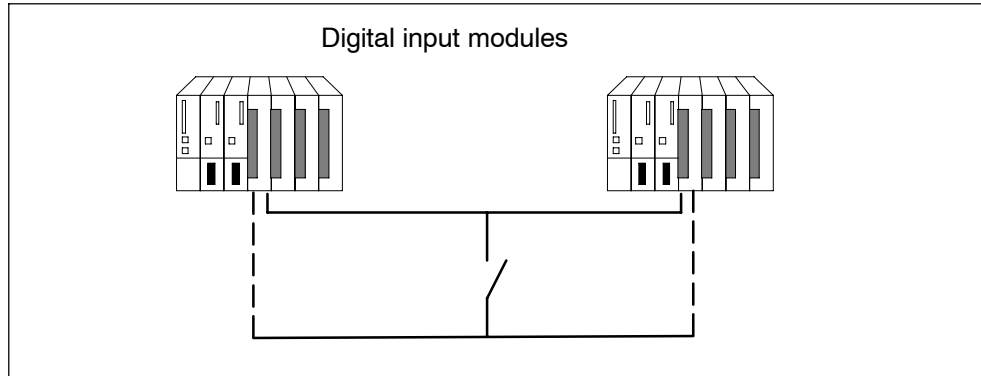
Figure 8-7   Redundant digital input module in a 1-out-of-2 configuration with one sensor

The use of redundant digital input modules increases their availability.

Discrepancy analysis detects "Continuous 1" and "Continuous 0" errors of the digital input modules. A continuous 1 error means the value 1 is continuous at the input, a continuous 0 error means that the input is continuously dead. This may be caused a short-circuit to L+ or M, for example.

The current flow of the chassis ground connection between the module and the encoder should be reduced to the possible minimum.

When connecting an encoder to several digital input modules, the redundant modules operate at the same reference potential.

Examples of these interconnections are found in appendix F .

Note

Note that current output value by the proximity switches (Beros) must be equivalent to twice the current specified in the technical data of the various modules.

Using redundant digital input modules with redundant sensors

You install digital input modules with redundant sensors in a 1-out-of-2 configuration:
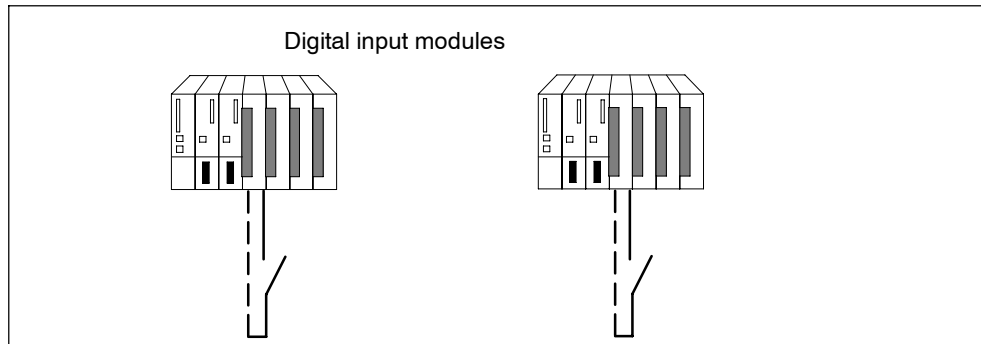
Figure 8-8   Redundant digital input modules in a 1-out-of-2 configuration with two

encoders

The redundancy of the encoders increases their availability. Discrepancy analysis detects all errors, except for the failure of a non-redundant load voltage supply. You can enhance availability by installing redundant load power supplies.

When connecting an encoder to several digital input modules, the redundant modules operate at the same reference potential.

Examples of these interconnections are found in appendix F.

Redundant digital output modules

Redundant control of an actuator can be achieved by connecting two outputs of two digital output modules or fail-safe digital output modules in parallel (1-out-of-2 configuration)
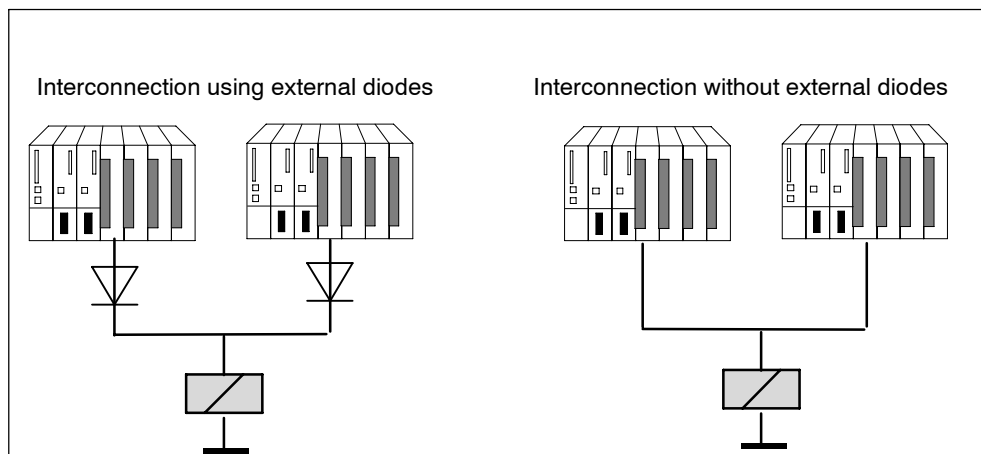
Figure 8-9   Redundant digital output module in a 1-of-2 configuration

The digital output module must be connected to a common load voltage supply.

Examples of these interconnections are found in appendix F

Interconnection using external diodes <-> without external diodes

The table below lists the redundant digital output modules you should interconnect using external diodes according to 8-9:

Table 8-4   Interconnecting digital output module with/without diodes

Modules	with diodes	without diodes
6ES7 422-7BL00-0AB0	X	-
6ES7 422-1FH00-0AA0	-	X
6ES7 326-2BF01-0AB0	X	X
6ES7 322-1BL00-0AA0	X	-
6ES7 322-1BF01-0AA0	X	-
6ES7 322-8BF00-0AB0	X	X
6ES7 322-1FF01-0AA0	-	X
6ES7 322-8BH01-0AB0	-	X
6ES7 322-5SD00-0AB0	X	-

Information on wiring the diode circuit

- Suitable diodes are those of the series 1N4003 ... 1N4007, or any other diode with $V_r >= 200$ V and $I_F >= 1$ A

- It is advisable to separate chassis ground of the module and load ground. Both circuits should be interconnected to equipotential ground.

Using analog input modules as redundant I/O

Parameters you configured for the redundant operation of analog input modules:

- Tolerance window (configured as a percentage of the end value of the measuring range)
  Two analog values are considered equal if they are within the tolerance window.

- Discrepancy time (maximum time in which the redundant input signal can be outside the tolerance window)
  An error is generated when there is an input value discrepancy after expiration of the configured discrepancy time.
  If you connect identical sensors to both analog input modules, the default value for the discrepancy time is usually sufficient. If you connect different sensors, in particular temperature sensors, you may have to increase the discrepancy time.

- Applied value
  The applied value represents the value of the two analog input values that is entered into the user program.

The system verifies that the two read analog values are within the configured tolerance window. If yes, the applied value is written to the lower data memory area of the process image of inputs. If there is a discrepancy and it is the first discrepancy, it is marked and the discrepancy time is started.

When the discrepancy time is running the most recently valid value is written to the process image of the module with the lower address and made available to the current process. When the discrepancy time is expired the module/channel with the configured standard value is declared as valid and the other **module/channel** is passivated. If the maximum value from both modules is configured as the standard value, this value is then taken for further program execution and the other **module/channel** is passivated. If the minimum value is set, this module supplies the data to the process and the module with the maximum value is passivated. The passivated module/channel is registered in the diagnostic buffer in any case.

When the discrepancy ceases within discrepancy time, the analysis of the redundant input signals continues.

Notice

The time that the system actually needs to determine a discrepancy depends on several factors: bus transit times, cycle and call times in the user program, conversion times etc. Redundant input signals may therefore be longer than the difference in the configured discrepancy time.

Note

The discrepancy analysis will be discarded when a channel reports an overflow with 16#7FFF, or an underflow with 16#8000, and the relevant module/channel will be passivated immediately.

You should thus disable all unused inputs in HW Configat the "measurement type"parameter.

Redundant analog input modules with non-redundant encoders

Analog input modules in a 1-out-of-2 configuration are operated with non-redundant encoders:

| Analog input modules | Analog input modules | Analog input modules |

Voltage measurement   Indirect current measurement   Direct current measurement
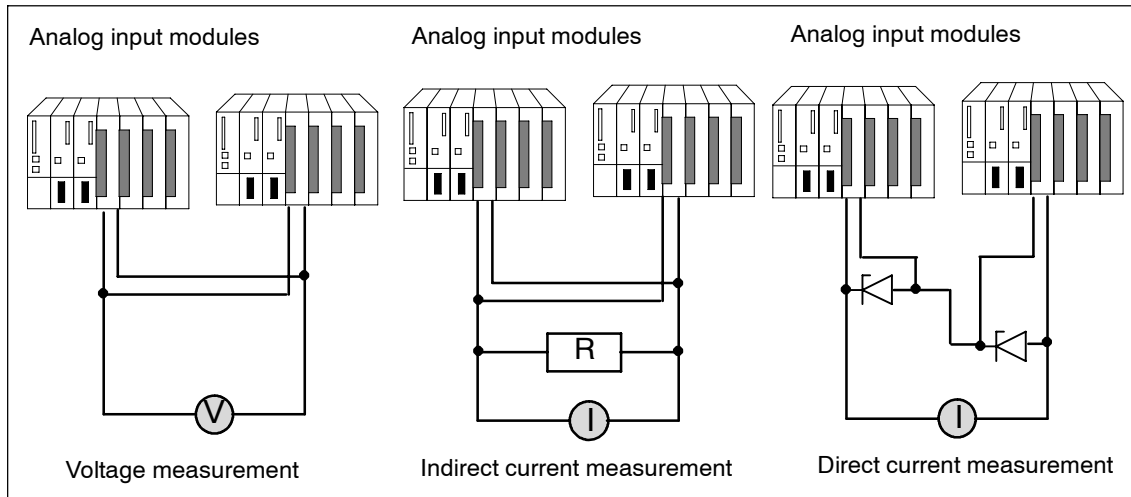
Figure 8-10   Redundant analog input modules in a 1-out-of-2 configuration with one encoder

Pay attention to the following when connecting a sensor to several analog input modules:

- Connect the voltage sensors in parallel to the analog input modules (left in illustration).

- You can transform a current into voltage using an external load to use voltage analog input modules connected in parallel (center in the illustration.)

- 2-wire measuring transducers are powered externally to enable you to repair the module online.

The use of fail-safe analog input modules enhances availability.

Examples of these interconnections are found in appendix F.

Redundant analog input modules for indirect current measurements

Requirements of circuits with analog input modules according to Figure 8-10:

- Suitable encoders forthis circuit: active measurement transducers with voltage output, thermocouples.

- Always disable the "wire break" diagnostics function in HW Config when operating the modules with measuring transducers or thermocouples.

- Suitable sensor types are active 4-wire and passive 2-wire-measuring transducers with output ranges +/-20 mA, 0...20 mA and 4...20 mA. 2-wire-measuring transducers are supplied with power by an external auxiliary voltage.

- Criteria for the selection of resistance and input voltage range are the measurement precision, number format, maximum resolution and diagnostics.

- In addition to the options listed, other input resistance and voltage combinations according to Ohm's law are also possible. Note that such combinations may lead to the loss of the number format, diagnostics function and resolution. The measurement error is also highly dependent on the value of the shunt resistance for certain modules.

- Use a measurement resistance with a tolerance of +/- 0.1% and TC 15 ppm.

Further marginal requirements of specific modules

AI 8x12Bit   6ES7 331-7K..02-0AB0

- Use a 50 Ohm or 250 Ohm resistance to map the current to a voltage:

Resistance	50 Ohm	250 Ohm	
Current measuring range	+/-20 mA	+/-20 mA *)	4...20 mA
Input range to be configured	+/-1 V	+/-5 V	1...5V
Measuring range cube position	"A"	"B"	
Resolution	12 bit + sign	12 bit + sign	12 bit
S7 number format	x	x	

Circuit-specific measuring error
  - 2 parallel inputs
  - 1 input
	
	-
	-
		
		0.5%
		0.25%	

Diagnostics "wire break"	-	-	x *)
Load for 4-wire measuring transducer	50 Ohm	250 Ohm	

Resistance	50 Ohm	250 Ohm
Input voltage of 2-wire measuring transducer	> 1.2 V	> 6 V

*) The AI 8x12bit outputs diagnostic interrupts and measuring value "7FFF" in the event of wire break.

The listed measuring error is derived alone from the interconnection of one or two voltage inputs with a shunt resistance. Allowances have not been made here for the error tolerance, or for the basic / operational limits of the modules.

The measuring error at one or two inputs shows the difference in the result of the measurement, depending whether two inputs, or one input in error case, register the current of the measuring transducer.

AI 8x16bit   6ES7 331-7NF00-0AB0

• Use a 250 Ohm resistance to map the current to a voltage. See page

Resistance	250 Ohm *)	
Current measuring range	+/-20 mA	4...20 mA
input range to be configured	+/-5 V	1...5 V
Resolution	15 bits + sign	15 bits
S7 number format	x	

Circuit-specific measuring error
  - 2 parallel inputs
  - 1 input
	-
	-	

Diagnostics "wire break"	-	x
Load for 4-wire measuring transducer	250 Ohm	
Input voltage of 2-wire measuring transducers	> 6 V	

*) you may also wire up the internal 250 Ohm resistor circuit of the module

AI 16x16bit   6ES7 431-7QH00-0AB0

• Use a 50 Ohm or 250 Ohm resistance to map the current to a voltage:

Resistance	50 Ohm	250 Ohm	
Current measuring range	+/-20 mA	+/-20 mA	4...20 mA
input range to be configured	+/-1 V	+/-5 V	1...5 V
Measuring range cube position	"A"	"A"	
Resolution	15 bits + sign	15 bits + sign	15 bits
S7 number format	x	x	

Switching cond. meas. error 1)
  - 2 parallel inputs
  - 1 input
	-
	-
		-
		-	

Diagnostics "wire break"	-	-	x

Resistance	50 Ohm	250 Ohm
Load for 4-wire measuring transducer	50 Ohm	250 Ohm
Input voltage of 2-wire measuring transducers	> 1.2 V	> 6 V

Redundant analog input modules for direct current measurements

Requirements of circuits with analog input modules according to Figure 8-10 :

- Suitable sensor types are active 4-wire and passive 2-wire-measuring transducers with output ranges +/-20 mA, 0...20 mA and 4...20 mA. 2-wire-measuring transducers are supplied with power by an external auxiliary voltage.

- The "wire break" diagnostics function supports only the 4...20 mA input range. All other unipolar or bipolar ranges are here excluded.

- You can use any diode of the BZX85 or 1N47..A (Zener diodes 1.3 W) series, provided these are suitable for the voltages specified for the modules. The reverse locking current of any other elements you may select should be as low as possible.

- A basic measuring error develops with this type of circuit containing the specified diodes, due to the maximum reverse locking current of $1\mu A$. In the 20 mA range, and at a resolution of 16 bits, this leads to an error of < 2 bits. Various analog input operations in the circuit shown above may lead to an additional error which may be specified in the marginal conditions. These errors, plus those specified in the manual, form the accumulated error rate at all modules.

- The 4-wire measuring transducers must be capable of driving the load resistance derived from the circuit shown above.. Details are found in the technical data of the various modules.

- When using 2-wire measuring transducers, you should be aware that the Zener diode circuit represents a heavy load on the load supply of the measuring transducers. The required input voltages are therefore included in the technical data of the various modules. The inherent supply specified in the data sheet of the measuring transducer plus is included in the calculation of the minimum supply voltage to $L+ > V_{in-2Dr} + V_{IV-MV}$

Redundant analog input modules with redundant encoders

In a 1-out-of-2 configuration, fail-safe analog input modules are given the preference for operation with double redundant sensor:
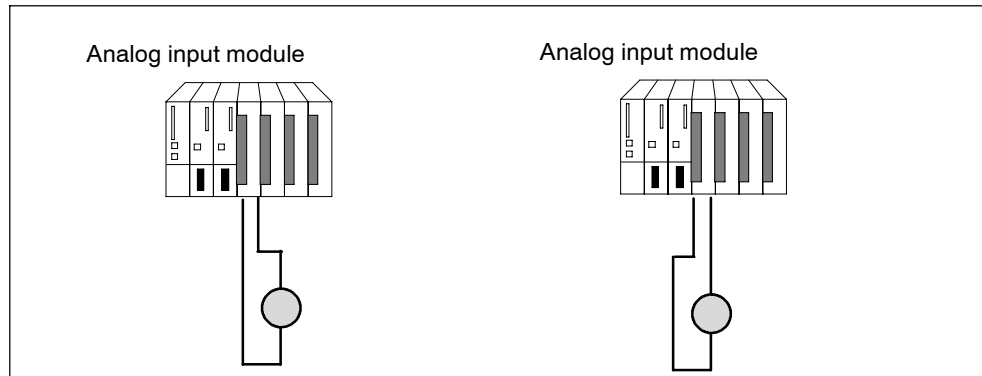
Figure 8-11   Redundant analog input modules in a 1-out-of-2 configuration with two
                encoders

The use of redundant sensors enhances their availability.

A discrepancy analysis also detects external errors, except for the failure of a non-redundant load voltage supply.

Examples of such interconnections are found in appendix F.

The initial information provided in this documentation applies

Redundant encoders <-> non-redundant encoders

The table below shows you which analog input modules you may operate either with redundant or non-redundant modules:

Table 8-5   Analog input modules and encoders

Module	Redundant encoders	Non-redundant encoders
6ES7 431-7QH00-0AB0	X	X
6ES7 336-1HE00-0AB0	X	-
6ES7 331-7KF02-0AB0	X	X
6ES7 331-7NF00-0AB0	X	X
6ES7 331-7RD00-0AB0	X	X

Redundant analog output modules

You implement redundant control of an actuator by wiring two outputs of two analog output modules in parallel (1-of-2 structure)
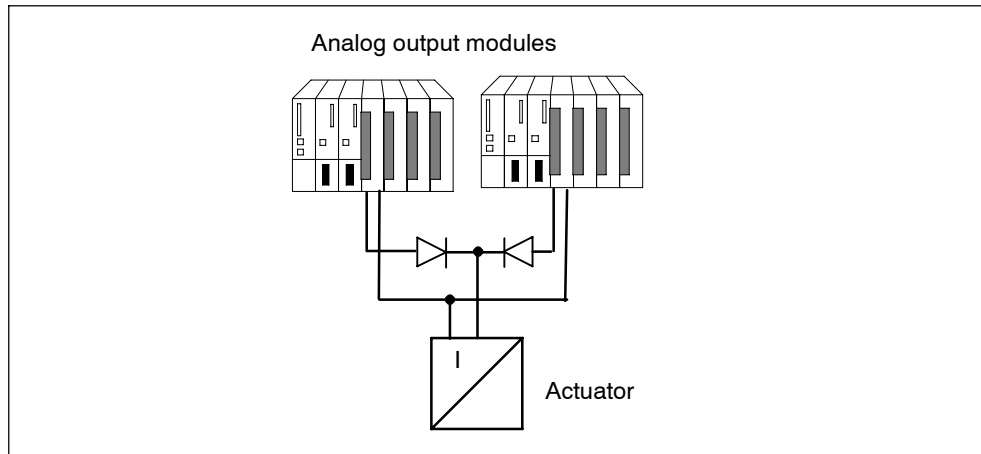
Figure 8-12   Redundant analog output modules in a 1-of-2 structure

Requirements of circuits with analog output modules according to Figure 8-12:

- Wire the ground connections in a star circuit in order to avoid output errors (limited common-mode suppression of the analog output module).

Information on the diode circuit

- Suitable diodes are those of the series 1N4003 ... 1N4007, or any other diode with $V_r >= 200$ V and $I_F >= 1$ A

- It is advisable to separate chassis ground of the module and load ground. Both circuits should be interconnected to equipotential ground.

Analog output signals

Redundant operation always requires analog output modules with current outputs (0 to 20 mA, 4 to 20 mA).

The value to be output is divided by 2, and the resultant values are output by the two modules. Hence, if a failure of one of the modules is detected, the partner module outputs the full value. The surge at the output module caused by this error is thus neglectable.

Notice

The output value drops briefly to 50%, and the program then recovers the proper value.

Redundant analog outputs provide a minimum current of approx. 120 μA per module, and thus an accumulated current of approx. 240 μA Hence, making allowances for tolerances, the module always outputs a positive value. A configured substitution value of 0mA will produce at least those output values. In redundant mode, the current outputs are automatically set to "off currentand off voltage".

Notice

If at two redundant analog output modules an error occurs on the second module, as long as the first module is passivated, the second cannot be passivated. If the first module is repaired and depassivated, only half the current value is output to the faulty channels until the second module is also repaired.

Depassivation of modules

Passivated modules are depassivated by the following events:

* When the redundant system starts up

* When the redundant system changes over to "redundant" state

    FB 452 "RED_DIAG" initiates depassivation at the transition to redundant operation. This requires a call of FB 452 in OB 72 (CPU redundancy error).

    FB 452 "RED_DIAG" also needs to be called in OB 82 (diagnostics interrupt), in OB 83 (removal/insertion interrupt) and in OB 85 (program runtime error). This ensures proper operation of the blocks for the redundant I/O.

* After system modifications in run

* If you call the FC 451 "RED DEPA" and, if at least one redundant channel or one redundant module is passivated. The functionality and use of FC 451 are described in the corresponding online help.

The depassivation is executed in FB 450 "RED IN" after any one of these events have occurred. Completion of the depassivation of all modules is logged in the diagnostics buffer.

When operating redundant I/O on a one-sided central device or one-sided DP slave, you have to depassivate the redundant modules after a station failure/recovery or replacement of a defective module. You can trigger a depassivation of all modules by calling FC 451.

Note

When a redundant module is assigned a process image partition and the corresponding OB is not available in the CPU, the complete passivation may take approximately 1 minute.

8.4.1 Evaluating the passivation status

Procedure

First, determine the passivation status by evaluating the status byte in the status/control word "FB_RED_IN.STATUS_CONTROL_W" . If you then see that a module was passivated, evaluate the status of all modules or module pairs in MODUL_STATUS_WORD.

Evaluating the passivation status using the status byte

The statusword "FB_RED_IN.STATUS_CONTROL_W" is located in the instance DB of FB 450 "RED_IN". The status byte returns information on the status of the redundant I/O.

Table 8-6   Assignment of the status byte

Bit	Meaning
Status byte (byte 1)	
0	Standby
1	In the case of module-granular redundancy: reserve In the case of channel-granular redundancy: 0 = no channel of the module is passivated 1 = at least one channel of the module is passivated
2	0 = no analog output module found 1 = at least one analog output module was found
3	0 = no passivation by OB 85 1 = at least one passivation by OB 85
4	0 = no passivation by OB 82 1 = at least one passivation by OB 82
5	0 = no channel information available 1 = at least channel information available
6	0 = no module passivated 1 = at least one module passivated
7	0 = complete depassivation not busy 1 = complete depassivation is busy

Evaluating the passivation status of individual modules by means of MODUL_STATUS_WORD

MODUL_STATUS_WORD is in located the instance DB of FB 453 "RED_STATUS". The two status bytes provide information about the status of individual module pairs. MODUL_STATUS_WORD is an output parameter of FB 453 and can be interconnected accordingly.

Table 8-7   Assignment of status bytes

Bit	Meaning
Status byte 1	
0	0 = Passivation of module-Low triggered by OB 82 1 = No passivation of module-Low triggered by OB 82
1	0 = Passivation of module-High triggered by OB 82 1 = No passivation of module-High triggered by OB 82
2	0 = Overflow or underflow (at analog input modules) 1 = No overflow or underflow
3	0 = Channel information is available 1 = Channel information is not available
4	0 = At least one discrepancy time expired (at input modules) 1 = No discrepancy time expired
5	0 = Module pair is discrepant (at input modules) 1 = Module pair is not discrepant
6	0 = Module-Low passivated 1 = Module-Low depassivated
7	0 = Module-High passivated 1 = Module-High depassivated
Status byte 2	
0	In the case of module-granular redundancy: reserve In the case of channel-granular redundancy: 0 = at least one channel of the module-Low is passivated 1 = no channel of the module-Low is passivated
1	In the case of module-granular redundancy: reserve In the case of channel-granular redundancy: 0 = at least one channel of the module-High is passivated 1 = no channel of the module-High is passivated
2	0 = No enable for depassivation of module-Low after outgoing event in the OB 85 1 = Enable for depassivation of module-Low after outgoing event in the OB 85
3	0 = No enable for depassivation of module-High after outgoing event in the OB 85 1 = Enable for depassivation of module-High after outgoing event in the OB 85
4	0 = No enable for depassivation of module-Low after outgoing event in the OB 82 1 = Enable for depassivation of module-Low after outgoing event in the OB 82
5	0 = No enable for depassivation of module-High after outgoing event in the OB 82 1 = Enable for depassivation of module-High after outgoing event in the OB 82
6	0 = Passivation of module-Low triggered in the OB 85 1 = No passivation of module-Low triggered in the OB 85
7	0 = Passivation of module-High triggered in the OB 85 1 = No passivation of module-High triggered in the OB 85

8.5 Other options of connecting redundant I/O

Redundant I/O on the user level

If you can not to use any of the redundant I/O supported by your system (chapter 8.4), because the relevant module may not be listed in the catalog of supported components, you could implement the redundant I/O on user level.

Configurations

The following redundant I/O configurations are supported (Figure 8-13):

1. Redundant configuration with one-sided central and/or distributed I/O.

   One I/O module is inserted for respectively in the CPU 0 and CPU 1 units.

2. Redundant configuration with switched I/O

   Two I/O modules are inserted into two ET 200M distributed I/O devices with an active backplane bus.
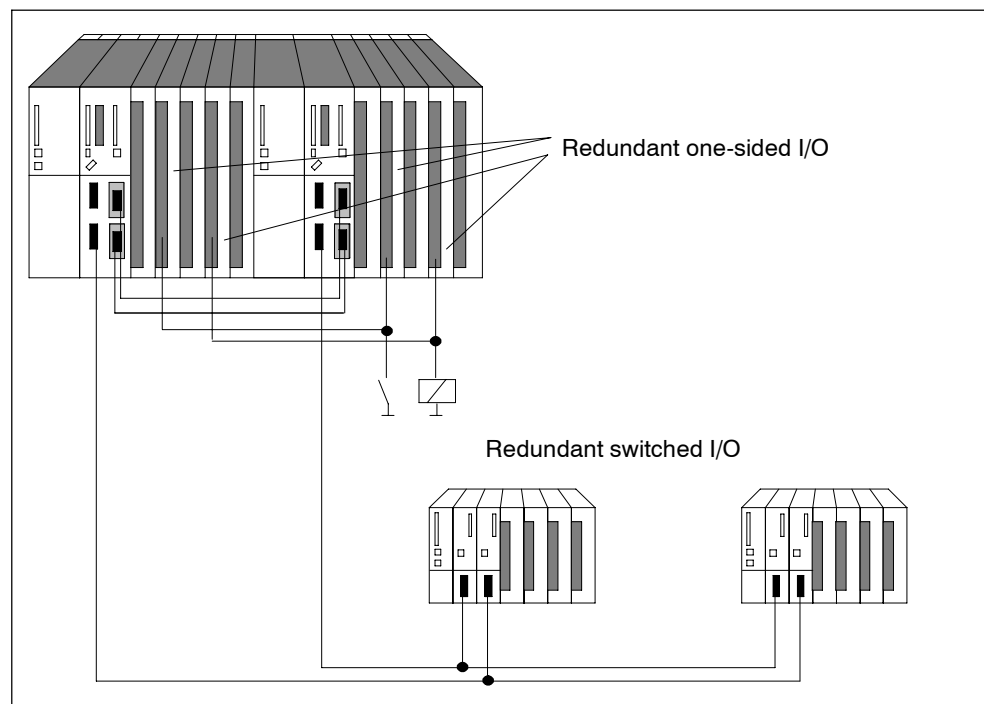
Figure 8-13   Redundant one-sided and switched I/Os

Notice

When using redundant I/O, you may have to add a premium to the calculated monitoring times; refer to Section 7.4.2.

HW installation and configuration of the redundant I/O

Strategy recommended for the use of redundant I/O:

1. Use the I/O as follows:

   – in a one-sided configuration, one I/O module per CPU unit

   – in a switched configuration, one I/O module per distributed I/O device ET 200M.

2. Wire the I/O in such a way that it can be addressed by both units.

3. Assign the I/O modules different logical addresses.

Notice

It is not advisable to assign the same logical addresses both to input and to output modules. When doing so 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 also has to update the process image for redundant one-sided output modules when the system is in stand-alone mode (direct access, for example). The user program in OB72 (recovery of redundancy) has to update all process image partitions (SFC27 "UPDAT_PO"), for the system would otherwise initialize the single-channel one-sided output modules of the standby CPU with the old values after the system transition to redundant state.

Redundant I/O in the user program

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

• module A in rack 0, logical base address 8 and

• module B in rack 1, logical base address 12.

One of the two modules is read in OB1 by direct access. Let us generally assume that module A is relevant, as the value of tag BGA is TRUE. If no error occurred, processing continues with the value read.

After an I/O access error has occurred, module B will be read by direct access ("retry" in OB1). Processing will otherwise continues with the value read by module B. However, if an error has similarly occurred in this instance, both modules are considered defective, and operations continues with a substitute value.

The example program is based on the fact that module B is always processed first in OB1 following an access error to module A or after the replacement of module A. Module A will not be processed first again in OB1 unless an access error occurs at module B.

Notice

The BGA and PZF_BIT tags must also be valid outside OB1 and OB122. The VERSUCH2 tag, however, is used only in OB1.
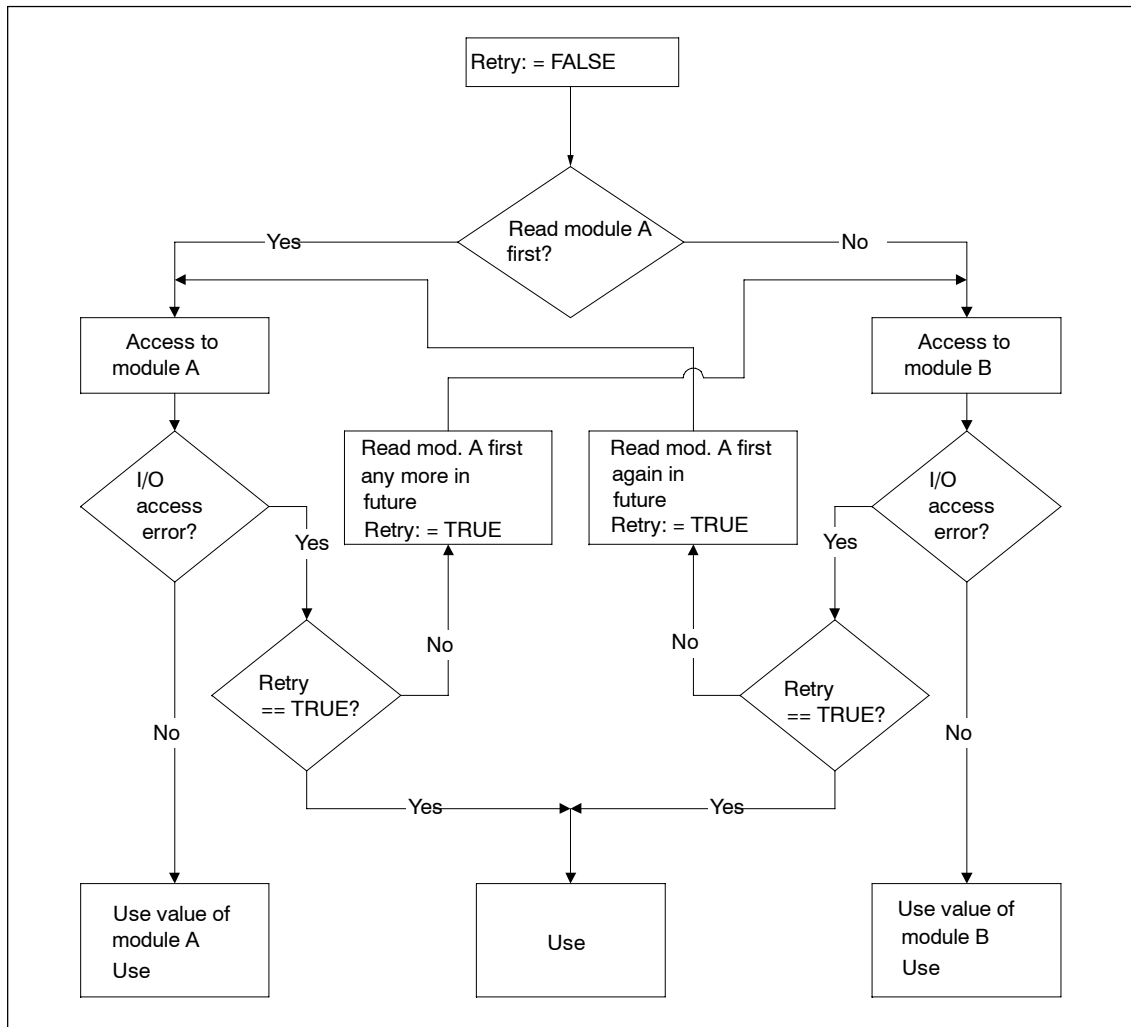
Figure 8-14   Flow chart for OB1

Example in STL

The required elements of the user program (OB1, OB 122) are listed below.

Table 8-8   Example of redundant I/O, OB1 part

STL	Explanation

NOP   0;
SET;
R   VERSUCH2;
A   BGA;
JCN WBGB;
	
	//Initialization
	//Read module A first?
	//If No, continue with module B

WBGA: SET;
R PZF_BIT;
L PED 8;
U PZF_BIT;
SPBN PZOK;
U VERSUCH2;
SPB WBG0;
SET;
R BGA;
 	
S VERSUCH2;
	
	//Clear process access error (PAE) bit
	//Read CPU 0
	//Was PAE detected in OB 122?
	//If NO, I/O access OK
	//Was this access the retry?
	//If yes, use substitution value
	
	//Do not read module A first any more
	//in future

WBGB: SET;
R PZF_BIT;
L PED 12;
U PZF_BIT;
SPBN PZOK;
U VERSUCH2;
SPB WBG0;
SET;
S   BGA;
S   VERSUCH2;
JU   WBGA;
	
	//Clear PAE bit
	//Read CPU 1
	//Was PAE detected in OB 122?
	//If no, process access OK
	//Was this access the retry?
	//If yes, use substitution value
	
	//Read module A first again in future

WBG0: L ERSATZ; PZOK:	//Substitution value //The value to be used is in Accu1

Table 8-9   Example of redundant I/O, OB1 part

STL	Description
	// Does module A cause PAE?
L OB122_MEM_ADDR;	//Relevant logical base address
L W#16#8;	
== I;	//Module A?
SPBN M01;	//If no, continue with M01
SET;	**//PAE during access to module A**
= PZF_BIT;	//Set PAE bit
SPA CONT;	
	// Does module B cause a PAE?
M01: NOP 0;	
L OB122_MEM_ADDR;	//Relevant logical base address
L W#16#C;	
== I;	//Module B?
SPBN CONT;	//If no, continue with CONT
SET;	**//PAE during access to module B**
= PZF_BIT;	//Set PAE bit
CONT: NOP 0;	

Monitoring periods on coupling and synchronizing

Notice

If you have made I/O modules redundant and have taken account of this accordingly in your program, you might have to add to the determined monitoring period so that no surges occur at output modules.

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

Table 8-10   Premium for the monitoring periods in the case of redundant I/O

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

You proceed as follows:

- You determine the premium from the table. If you have used a number of module types in the table redundantly, the largest premium is to be taken.

- Add this to all of the monitoring times determined so far.

Communication Functions

9

This chapter introduces communications with redundant systems and their specific characteristics.

It shows you the basic concepts, the bus systems you can use for redundant communications, and the available types of connection.

It contains information on communication functions using redundant and standard connections, and explains how to configure and program these.

- The document also deals with the advantages of communication by means of **redundant S7 connections**.

- By way of comparison, you will learn how communication takes place over **S7 connections** and also how you can communicate in redundant mode by means of S7 connections.

In Section	Description	On Page
9.1	Fundamentals and basic concepts	9-2
9.2	Suitable networks	9-5
9.3	Supported communication services	9-5
9.4	Communications by means of redundant S7 connections	9-6
9.5	Communications via S7 connections	9-13
9.6	Communication performance	9-19
9.7	General issues in communication	9-21

9.1 Fundamentals and basic concepts

Overview

Rising demands on the availability of an overall system make it inevitable to increase fail–safety of communication systems, i.e. communication systems must be designed for redundant operation.

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

Redundant communication system

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

Upon failure of a component, the various monitoring and synchronization mechanisms ensure that the communication functions are assumed by the standby components while the system stays in run.

A redundant communications system is prerequisite for the configuration of redundant S7 connections.

Redundant communications

The term redundant communications refers to the use of SFBs in S7 communications by means of redundant S7 connections.

Redundant S7 connections are only available within redundant communications systems.

Redundancy nodes

Redundancy nodes represent the fail–safety of communications between two redundant systems. A system with multi-channel components is represented by redundancy nodes. The independence of a redundancy 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 represents the logical assignment of two communication partners executing 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 redundant system.

In contrast to a redundant 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.
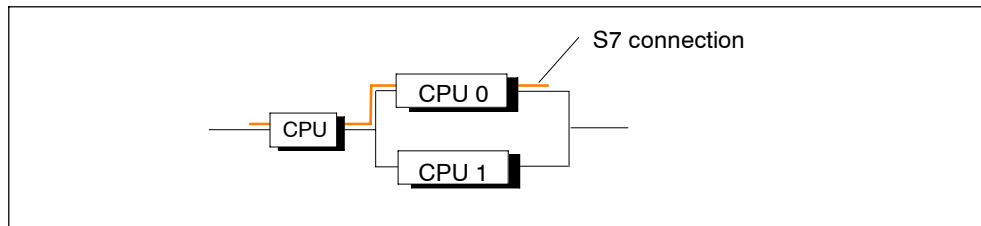
Figure 9-1   Example of an S7 connection

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*.

Redundant 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 redundant 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 redundant 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 9-2) and is determined automatically.
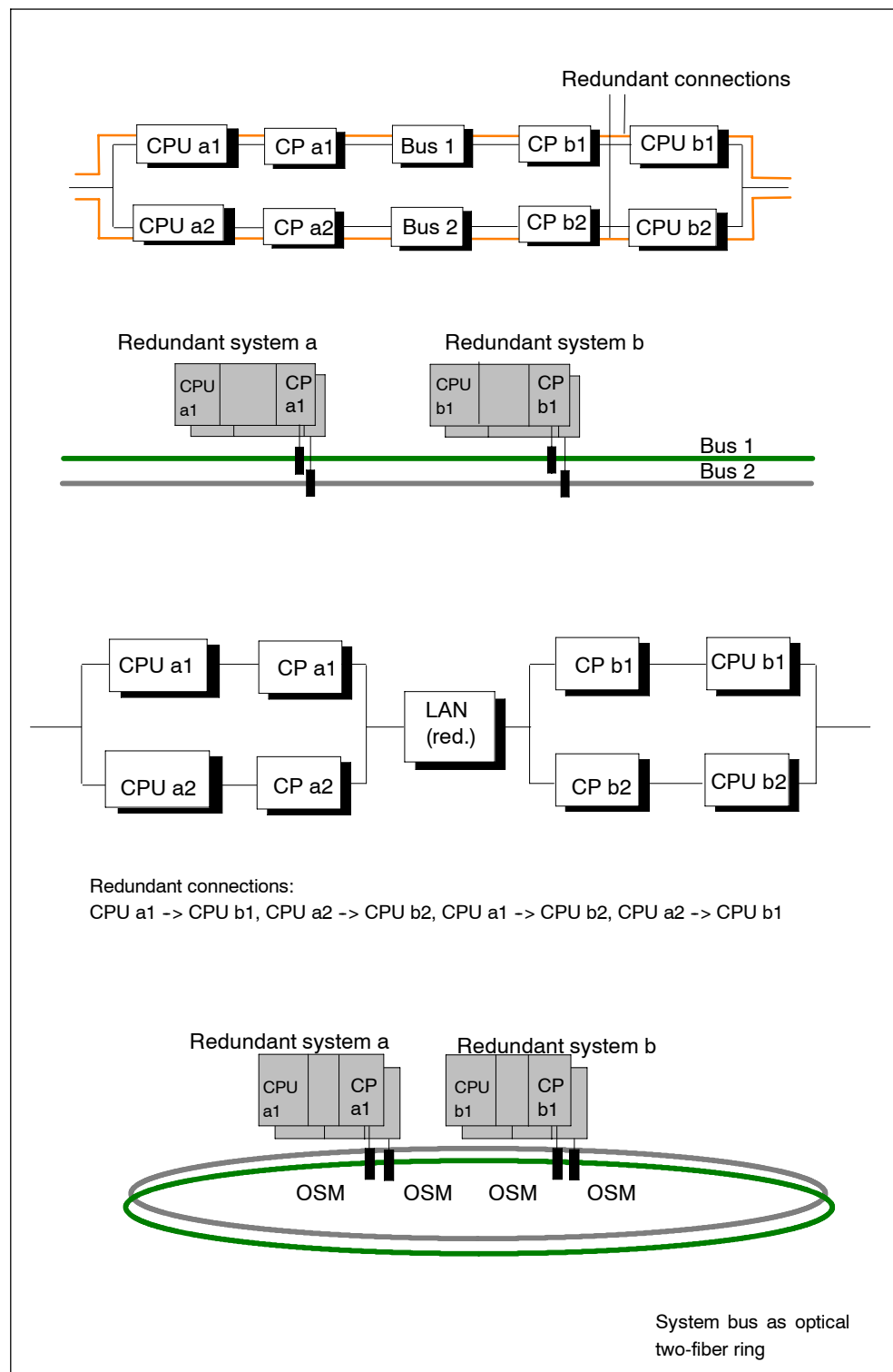
Figure 9-2   Example of the number of resulting partial connections being dependent on the configuration

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

Resource requirements of redundant S7 connections

The redundant CPU supports the operation of 62/30 (see the technical data) redundant S7 connections. On the CP each partial connection requires a connection resource.

Note

If you have configured several redundant S7 connections for a redundant station, setting them up may take a considerable length of time. If the configured maximum communication delay is too short, coupling and updating is canceled and the system status no longer reaches redundant state (see chapter 7.4).

9.2 Suitable networks

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

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

*   PROFIBUS (fiber-optic cable or copper cable)

For further information on suitable networks, refer to the "*Communication with SIMATIC*", *"Industrial Twisted Pair Networks"* and *"PROFIBUS Networks*" manuals.

9.3 Supported communication services

The following services can be used:

*   S7 communications using redundant S7 connections via PROFIBUS and Industrial Ethernet

*   S7 communications using 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:

*   S7 basic communications

*   Global data communication

*   Open Communication via Industrial Ethernet

9.4   Communications via redundant S7 connections

Availability of communicating systems

Redundant communications expand the overall SIMATIC system by additional, redundant communication components, such as CPs and LAN cables. 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.

Prerequisite

Prerequisite for the configuration of redundant connections with STEP 7 is a configured hardware installation.

The hardware configuration in both units of the redundant systems **must** be identical. This applies in particular to the slots.

Depending on the network being used, the following CPs can be used for redundant communications:

- Industrial Ethernet:
  S7: CP 443-1 EX10/EX11
  the system only supports Industrial Ethernet with ISO protocol.

To be able to use redundant S7 connections between a redundant system and a PC, you must install the "S7-REDCONNECT" software package on the PC. Please refer to the Product Information 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 redundant S7 connections. To set this up, open the "New Connection" dialog box, then select "S7 Connection Fault-Tolerant" as the type.

The number of required redundant connections is determined by STEP 7 as a function of the redundancy nodes. Up to four redundant connections will be generated, if supported by the network. Higher redundancy can not be achieved by using more CPs.

In the "Properties - Connection" dialog box, you can modify specific properties of a redundant connection, should you require to do so. When using more than one CP, you can also route the connections in this dialog box. This may be practical, because by default all connections are routed initially through the first CP. If all the connections are in use there, any further connections are routed through the second CP etc.

Programming

Redundant communication can be implemented on the redundant CPU and takes place by means of S7 communication.

This is possible solely within an S7 project/multiproject.

Redundant communications are programmed in STEP 7 by means of communication SFBs. Those blocks can be used to transfer data on subnets (Industrial Ethernet, PROFIBUS). The standard communication SFBs integrated in the operating system offer you the option of acknowledged data transfer. In addition to data transfer, you can also use other communication functions for controlling and monitoring the communication partner.

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

Note

For information on programming the communication, refer to the STEP 7 documentation (*Programming with STEP 7*, for example).

The START and STOP communication functions act on exactly one CPU or on all CPUs of the redundant system (for more details, refer to the *System Software for S7-300/400, System and Standard Functions* reference manual).

Any interruption of partial connections while communication requests via redundant S7 connections are busy will lead to extended runtimes.

9.4.1 Communications between Fault-Tolerant Systems

Availability

The easiest way to enhance availability between coupled systems is to implement a redundant system bus, using a multimode fiber optic ring or a dual electrical bus system. In this case the connected nodes may consist of simple standard components.

Availability can is enhanced ideally using a duplex fiber–optic ring structure. Should the one of the multimode fiber–optic cables break, communications continue to exist between the systems involved. The systems then communicate as if they were connected to a bus system (line). A ring topology basically contains two redundant components, and thus automatically forms a 1-of-2 redundancy node. A fiber optic network can be set up as a line or star topology. However, the line topology does not offer cable redundancy.

Should one electrical cable segment fail, communications between the partner systems is also upheld (1-of-2 redundancy).

The examples below illustrate the differences between both versions.

Note

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

If you implement a multimode fiber optic ring (refer to Figure 9-3), 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 9-4) is being used.

Figure 9-3   Example of redundancy with redundant system and redundant ring

Figure 9-4   Example of redundancy with redundant system and redundant bus system

Figure 9-5   Example of a redundant system with additional CP redundancy

Reaction to failure

Only a double error within a redundant system (for example, with CPUa1 and CPa2) with multimode optic fiber ring leads to total failure of communications between the redundant systems concerned (refer to Figure 9-3).

If a double error (CPUa1 and CPb2, for example) occurs in the first case of a redundant electrical bus system (see Figure 9-4), this results in a complete failure of communication between the systems involved.

In the case of a redundant electrical bus system with CP redundancy (see Figure9-5), only a double error within a redundant system (CPUa1 and CPUa2, for example) or a triple error (CPUa1, CPa22 and bus2, for example) will result in a complete failure of communication between the systems involved.

Redundant S7 Connections

Any interruption of partial connections while communication requests via redundant S7 connections are busy will lead to extended runtimes.

9.4.2 Communications between redundant systems and a redundant CPU

Availability

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

If the communication partner is a redundant H-CPU redundant connections may also be configured, in contrast to systems with 416 CPU, for example.

Note

Redundant connections use two connection resources on CP b1 for the redundant connections. One connection resource each is per CP a1 and CP a2.
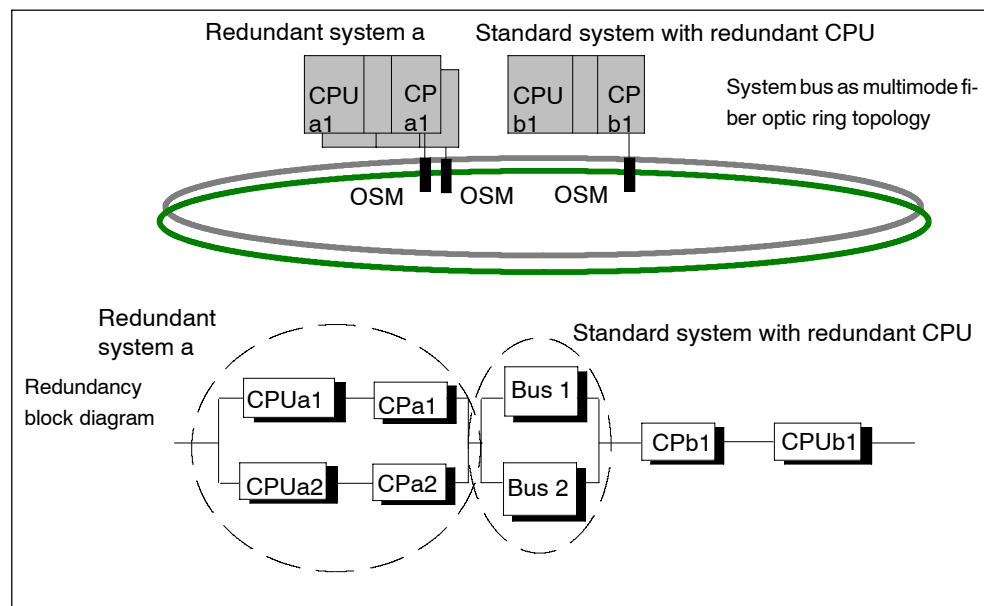
Figure 9-6   Example of redundancy with redundant system and redundant H–CPU

Reaction to failure

Double errors in redundant systems (i.e. CPUa1 and CPa2) and single errors in a standard system (CPUb1) will cause a total loss of communication between the relevant systems. See Figure 9-6

9.4.3 Communications between redundant systems and PCs

Availability

When redundant systems are coupled 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 automation systems.

PCs are not redundant on account of their hardware and software characteristics. They can be arranged in a redundant manner in the 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 redundant connections.

The "S7-REDCONNECT" software package, V1.3 or higher, is prerequisite for redundant communication on a PC. It supports the connection of a PC to an fiber optic network with one CP, or to a redundant bus system with 2 CPs.

Configuring connections

No additional configuration of redundant 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 redundant S7 communications to integrate a PC in your OS system in the WinCC documentation.
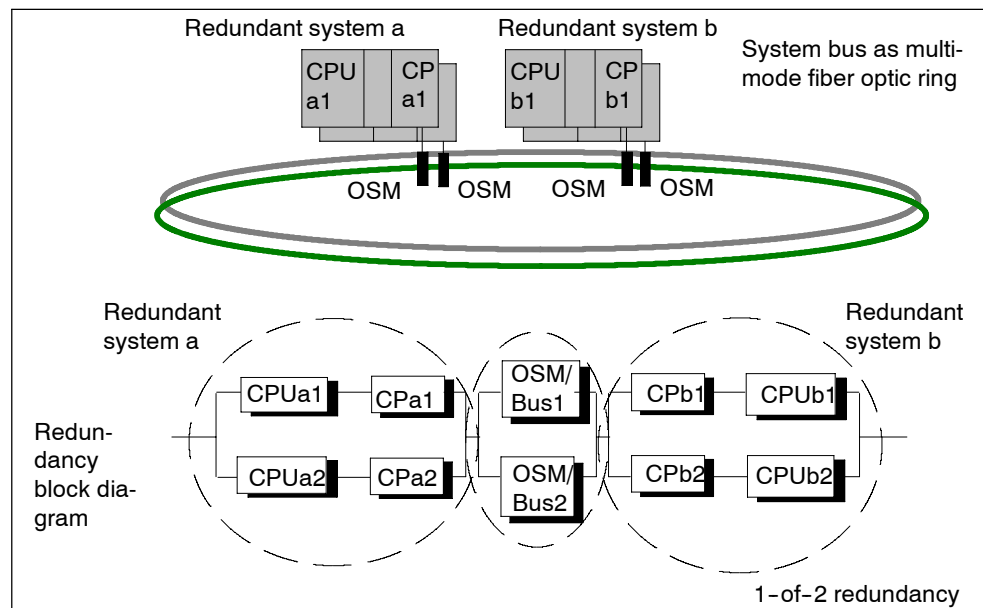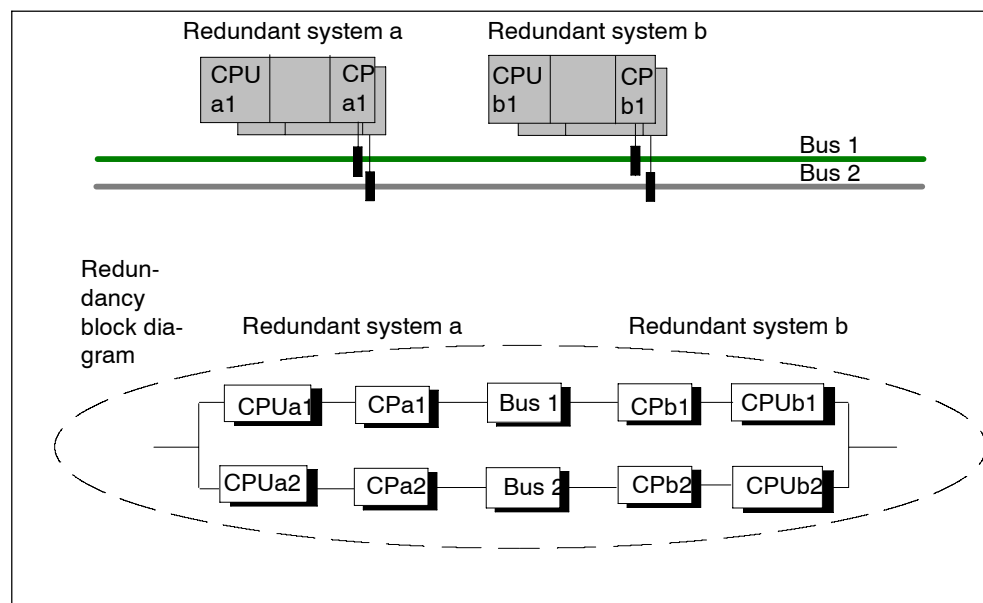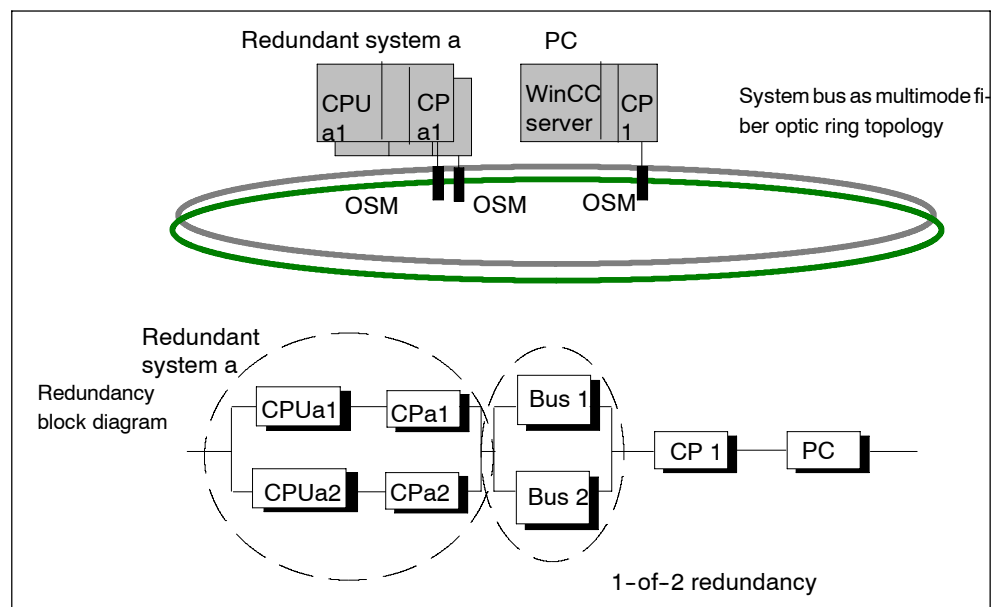

Figure 9-7   Example of redundancy with redundant system and redundant bus system

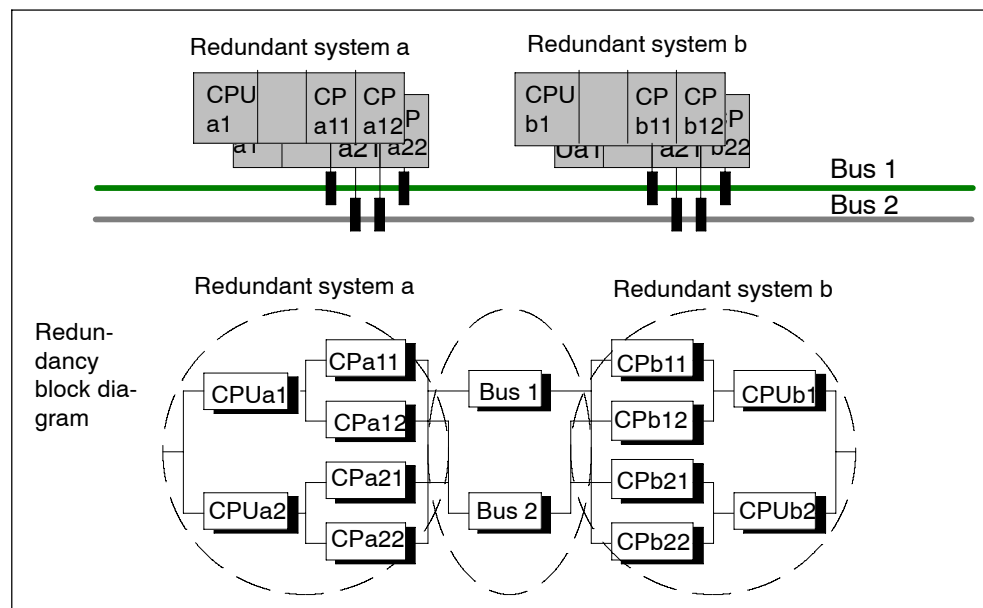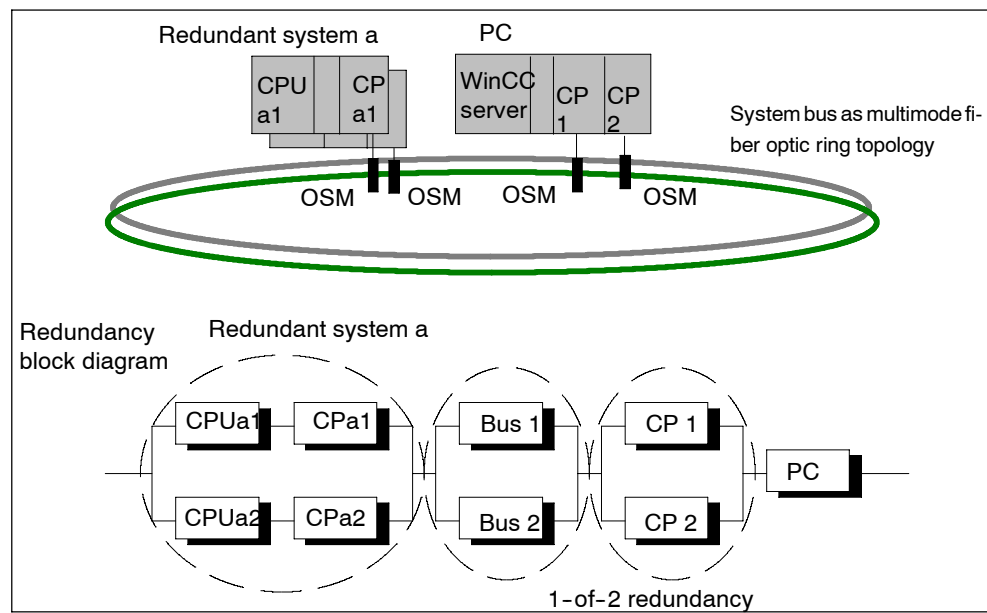Figure 9-8   Example of redundancy with a redundant system, redundant bus system, and CP redundancy in the PC

Reaction to failure

Double errors in redundant systems (i.e. CPUa1 and CPa2) and the failure of the PC result in a total loss of communication between the relevant systems (refer to Figures 9-7and 9-8).

PC / PG as Engineering System (ES)

In order to be able to use a PC as Engineering System, you need to configure it under its name as PC station in HW Config. The ES is assigned to a CPU and is capable of executing the STEP 7 functions on this CPU.

If this CPU fails, communication between the ES and the redundant system also goes down.

9.5 Communications via S7 connections

Communications with standard systems

Redundant communications between redundant and standard systems is not supported. The following examples illustrate the actual availability of the communicating systems.

Configuration

S7 connections are configure in STEP 7.

Programming

All communication functions except "global data communications" are supported for standard communications on a redundant system.

The standard communication SFBs are used in STEP 7 to program communications.

Note

The START and STOP communication functions act on exactly one CPU or on all CPUs of the redundant system (for more details, refer to the *System Software for S7-300/400, System and Standard Functions* reference manual).

9.5.1 Communications via S7 Connections – One-sided Connection

Availability

Availability is likewise enhanced by using a redundant system bus for communications between redundant and standard systems.

On a system bus configured as multimode fiber optic ring, communications between the partner systems goes down if the multimode fiber optic cables breaks. The systems then communicate as if they were connected to a (line) bus system. See Figure 9-9.

For coupled redundant and standard systems, the availability of communications can not be improved by means of a dual electrical bus system. In order to be able to use the second bus system as redundancy node, you need to configure a standby S7 connection and manage these accordingly in the user program (see Figure 9-10).
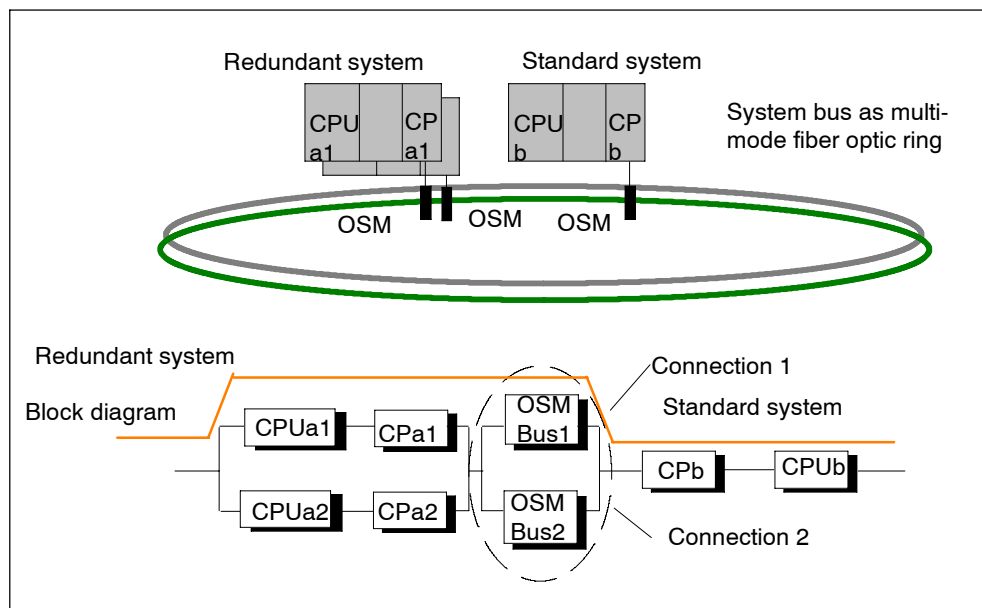
Figure 9-9   Example of the coupling between standard and redundant systems on a redundant ring
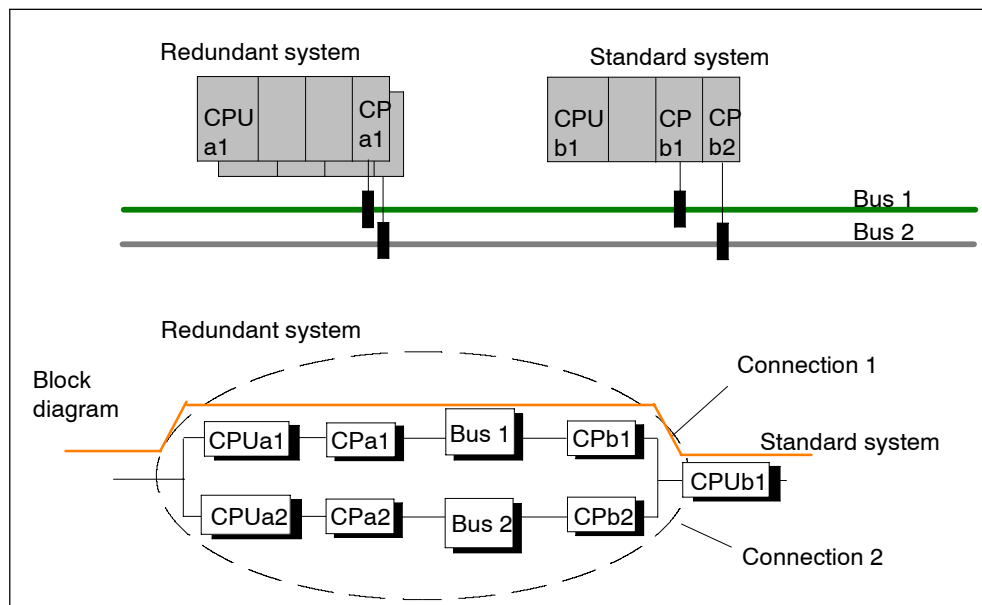
Figure 9-10   Example of the coupling between standard and redundant systems on a redundant ring

Automation System S7-400H Fault-tolerant Systems
A5E00267695-03

Reaction to failure

Multimode fiber optic ring and bus system

Because 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 redundant system (for example, CPUa1 or CPa1) or an error on system b (for example, CP b) will result in total failure of communication between those partner systems (refer to Figures 9-9 and 9-10).

There are no differences in bus system–specific reactions to failure.

Coupling standard and H systems

Driver module "S7H4_BSR": To couple an H system with an S7-400, you can use the driver module "S7H4_BSR" from the STEP7 library. You can order this at:

http://www.khe.siemens.de/it/index1360712_1.htm

Alternatively, SFB 15 "PUT" and SFB 14 "GET" in the H system: As an alternative, use two SFB 15 "PUT" via two standard connections. First of all, start the first module. If there was no error message on running the module, the transfer is regarded as successful. If there was an error message, the data transfer is repeated across the second module. In the event of a connection abort that is detected later, the data is also transferred again to exclude information losses. You can use the same method for an SFB 14 "GET".

If possible, use the mechanisms of the S7 communication for communication.

9.5.2   Communications via redundant S7 Connections

Availability

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

Redundant communications may also be operated with standard connections. In this context, two separate S7 connections have to be configured in the program in order to implement connection redundancy. In the user program, both connections require the implementation of monitoring functions in order to allow the detection of failures and to changeover to the standby connection.

Figure 9-11 shows an example of such a configuration.
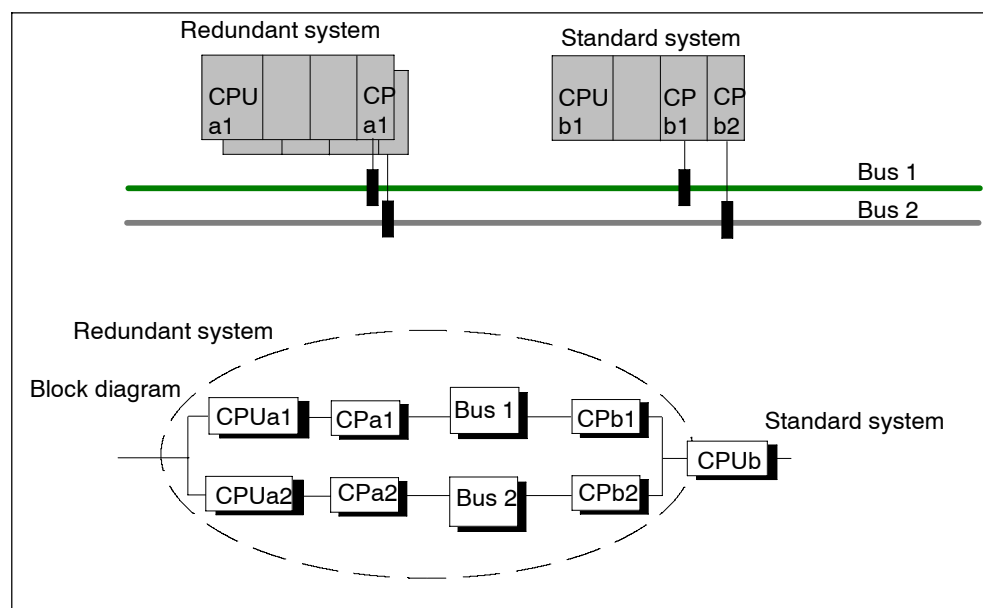
Figure 9-11   Example of redundancy with redundant systems, operating on a redundant bus system with redundant standard connections

Reaction to failure

Double faults in the redundant system (i.e. CPUa1 and CPa 2) or in the standard system (CPb1 and CPb2), and a single fault in the standard system (CPb1) will cause a total failure of communication between the redundant partners (refer to Figure 9-11).

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

Connection via ET200M

The connection of redundant systems to single-channel systems is usually possible only by way of PtP coupling, due to the lack of connection alternatives for certain systems.

In order to make the data of a single–channel system available to CPUs of the redundant systems as well, the PtP CP (CP 341) must be installed in a distributed rack alongside with two IM 153-2 modules.

Configuring connections

Redundant connections between the PtP CP and the redundant system are not necessary.
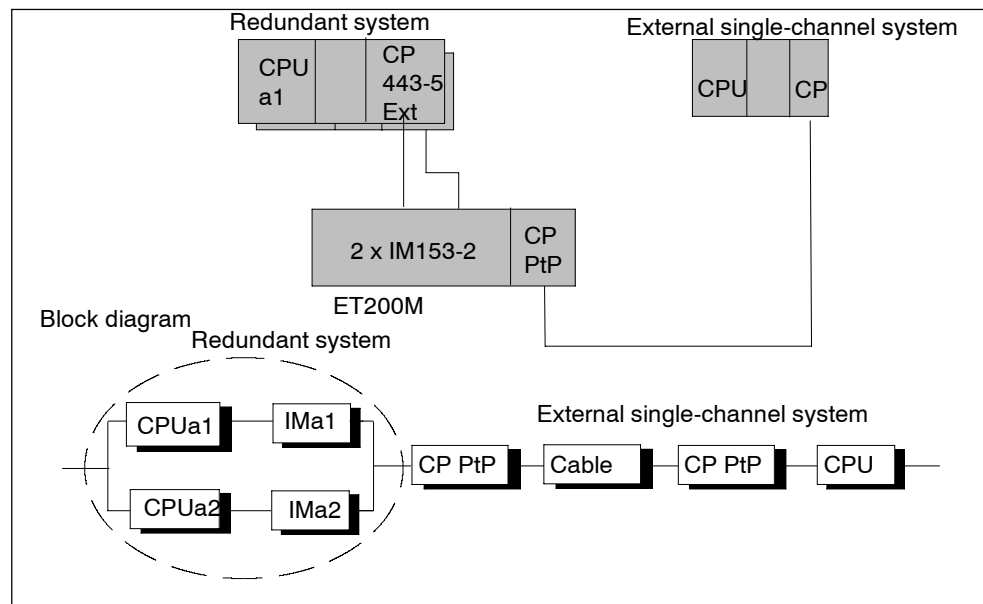
Figure 9-12   Example of the coupling of a redundant system and an external single-channel system

Reaction to failure

Double errors in the redundant system (i.e. CPUa1 and IM153-2) and single errors in the external system will cause a total failure of communication between the relevant systems (see Figure 9-12).

The PtP CP can alternatively be inserted centrally in "Redundant system a". However, in this configuration even the failure of a CPU will cause a total failure of communication, for example.

9.5.4 User–specific coupling with single-channel systems

Connection using the PC as gateway

Redundant and single-channel systems may also be coupled by means of gateway (no connection redundancy). The gateway is coupled to the system bus using one or two CPs, depending on availability requirements. You may configure redundant connections between the gateway and the redundant systems. The gateway thus allows you to couple any kind of single-channel system which is, for example, based on TCP/IP and a specific manufacturer's protocol).

A user–programmed software instance in the gateway implements the single-channel transition to the redundant systems, and thus allows the coupling of any single-channel system to a redundant system.

Configuring connections

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

The gateway CP is located on a PC system which has redundant connections to the redundant system.

To configure redundant S7 connections between the redundant system A and the gateway, you first need to install S7-REDCONNECT on the gateway. The functions for preparing data for their transfer via single–channel coupling must be implements in the user program.

For further information, refer to the "*Industrial Communications IK10*" catalog.
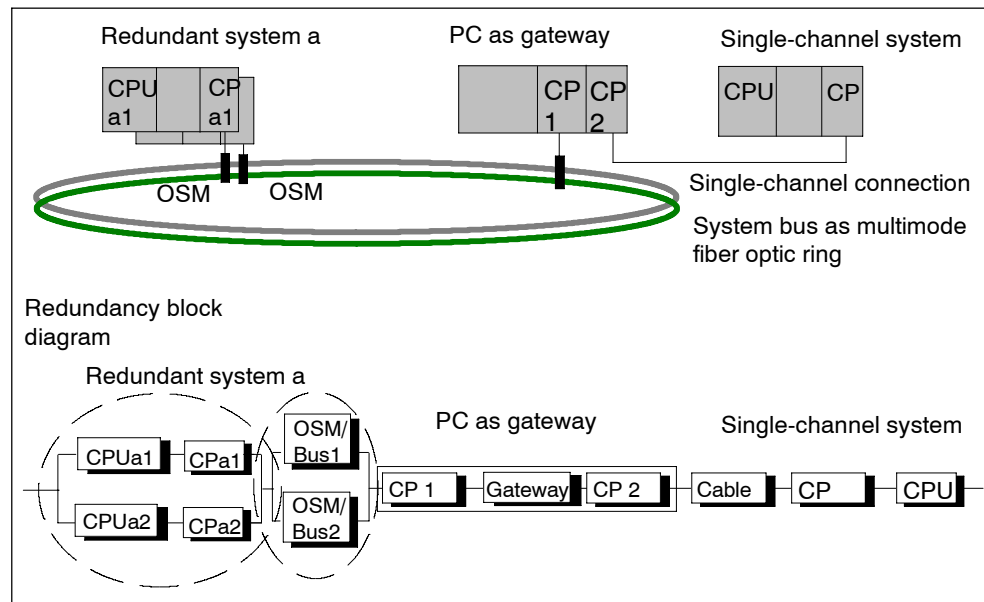
Figure 9-13   Example of the coupling of a redundant system and an external single-channel system

9.6 Communication performance

Compared to an H–CPU in stand–alone mode or to a standard CPU, the communication performance (reactions time or date thruput) an H system operating in redundant mode is significantly lower.

The objective of this manual is to provide you with certain rating criteria which allow you to assess the effects of the various communication mechanisms on communication performance.

Definition of communication load

Communication load is equivalent to the sum of requests output per second to the CPU by communication functions, including the requests and messages output by the CPU.

Higher communication load will increase reaction times of the CPU, i.e. the CPU requires more time to react to a request (read request, for example) or output messages.

Working range

Within the linear working range of the automation system, an increase in communication load will also lead to an increase in data thruput, and to manageable reaction times which are usually acceptable for the relevant automation application.

A further increase in communication load will push data thruput into the saturation range. Under certain conditions, the automation system may thus no longer be capable of processing the request volume within the response time demanded, data thruput reaches its maximum, and the reaction time rises exponentially. See the figures below.

Data thruput may also be reduced on the device by a certain amount due to additional internal loads.
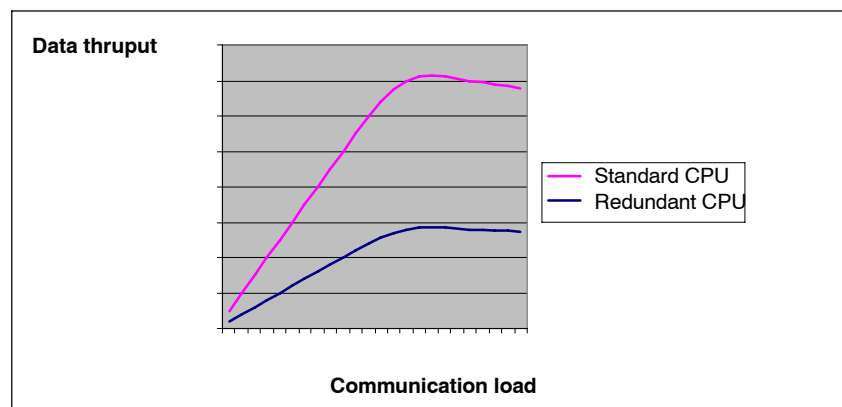
Figure 9-14   Communication load as a variable of data thruput (basic profile)
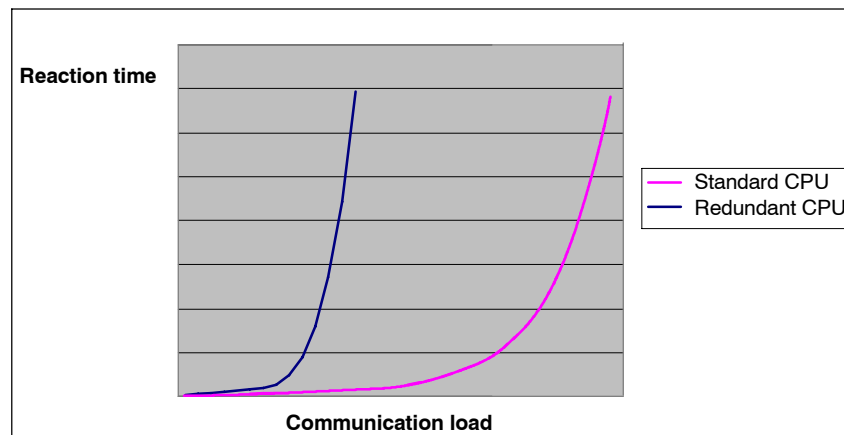
Figure 9-15   Communication load as a variable of reaction times (basic profile)

Standard and redundant systems

The information provided up to this point applies to standard and redundant systems. Considering that communication performance in standard systems is clearly higher compared to systems operating in redundant mode, we can say that state–of–the–art systems will only go into saturation in exceptional situations.

In contrast, redundant systems always require synchronization in order to uphold synchronism. This inevitably increases block runtimes and reduces communication performance, and thus leads to lower performance limits. If the redundant system is not operating at its performance limits, the performance benchmark compared to the standard system will be lower by the factor 2 to 3.

Variables having an influence on communication load

Communication load is influenced by the following variables:

- Number of connections/connected O&M systems

- Number of tags, or number of tags in images visualized on OPs or using WinCC.

- Communication type (O&M, S7 communication, S7 message functions, S5-compatible communication, ...)

- The configured cycle time extension as a result of communication load

The sections below shows the factors having an influence on communication performance.

9.7 General issues in communication

Reduce the rate of communication request per second as far as possible. Utilize the maximum user data length for communication requests, for example, by grouping several tags or data areas in one read request.

Each request requires a certain processing time, and its status should therefore not be checked before this process is completed.

You can download a tool for the assessment of processing times free of charge from the Internet under:

http://www4.ad.siemens.de/view/cs/de/1651770, contribution ID 1651770

Your calls of communication requests should allow the event–driven transfer of data. Check the data transfer event only until he request is completed.

Call the communication blocks sequentially and stepped down within the cycle, in order to obtain a balanced distribution of communication load.

You may by–pass the block call using a conditional jump if you do not transfer any user data.

A significant increase in communication performance between S7 components is achieved by using S7 communication functions, rather than S5-compatible communication functions.

As S5–compatible communication functions (FB "AG_SEND", FB "AG_RECV", AP_RED) generate a significantly higher communication load, you should only deploy these for the communication of S7 components with non–S7 components.

Software package AP-Red

When using the "AP_RED" software package, limit the user data length to 240 bytes. If larger data volumes are necessary, transfer those in sequential block calls.

The "AP_RED" software package uses the mechanisms of FB "AG_SEND" and FB "AG_RCV". Use AP_RED only to couple SIMATIC S5 / S5–H PLCs, or external devices which only support S5–compatible communication.

S7- communication (SFB 12 "BSEND" and SFB 13 "BRCV")

Do not call SFB 12 "BSEND" in the user program more often than the corresponding SFB 13 "BRCV" at the communication partner.

S7 communication (SFB 8 "USEND" and SFB 9 "URCV")

SFB 8 "USEND" should always be event–driven, because this block may generate a high communication load.

Do not call SFB 8 "USEND" in the user program more often than the corresponding SFB 9 "URCV" at the communication partner.

SIMATIC OPs, SIMATIC MPs

Do not install more than 4 OPs or 4 MPs in the redundant system. If you do need more OPs/MPs, your automation task may have to be revised. Contact your SIMATIC sales partner for corresponding support.

Do not select a screen update cycle time of less than 1s. Increase it to 2 s as required.

Verify that all screen tags are requested within the same cycle time, in order to form an optimized group for read requests.

OPC Server

When OPC was used to connect several HMIdevices for your visualization tasks to a redundant system, you should keep the number of OPCServers accessing the redundant system as low as possible. OPCclients should always address a shared OPCServer which will then fetch the data from the redundant system.

You can tune data exchange by using WinCC and its client/server concept.

Various HMI devices of third–party vendors support the S7communications protocol. You should utilize this option.

Configuring with STEP 7

<div style="text-align: right; font-size: 3em; font-weight: bold;">10</div>

This chapter provides an overview of fundamental issues to be observed when you configure a redundant system.

The second section covers the PG functions in STEP 7.

For detailed information, refer to *Configuring redundant systems* in the basic help.

10.1 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.

Notice

Always download these error OBs to the S7-400H CPU: OB 70, OB 72, OB 80, OB 82, OB 83, OB 85, OB 86, OB 87, OB 88, OB 121 and OB 122. If you ignore this, the redundant CPU goes into STOP when an error occurs.

Creating a redundant station

The SIMATIC H station represents a separate station type in SIMATIC Manager. It allows the configuration of two central controllers, each having a CPU and thus a redundant station configuration.

10.1.1 Rules for the assembly of redundant stations

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

- The CPUs always have to be inserted in the same slots.
- 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, rather than in expansion racks.
- Redundantly used modules (for example, CPU 417-4H, DP slave interface module IM 153-2) must be identical, i.e. they must have the same order number, the same version, and the same firmware version.

Installation rules

- A redundant 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 with communication bus interface 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 redundant 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 rack number is also assigned for DP master no. 9 onwards if the central rack contains 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.

10.1.2 Configuring Hardware

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

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

Special features in presenting the hardware configuration

In order to enable its quick recognition of a redundant DP master system, it is represented by two closely parallel DP cables.

10.1.3 Assigning parameters to modules in a redundant station

Introduction

Assigning parameters to modules in a redundant station is no different to assigning parameters to modules in S7-400 standard stations.

Procedure

All the parameters of the redundant components (with the exception of MPI and communication addresses) must be identical.

The special case of CPUs

You can only edit the CPU0 parameters (CPU on rack 0). Any values that you specify for it are automatically allocated to CPU1 (CPU on rack 1). The settings of CPU1 can not be changed, with the exception of the following parameters:

- MPI address of the CPU
- Integrated PROFIBUS DP properties

Configuring modules in the I/O address space

Always configure the access to modules in an I/O address space which lies either within or outside of the process image.

Otherwise, consistency can not be guaranteed, and the data may be corrupted.

I/O access using word or dword statements

The system loads the values to accumulator "0" if the word or dword for I/O access contains only the first or the first three bytes, but not the remaining bytes of the address space.

Example: The I/O is exists at address 8 and 9 in the S7-400H CPU, and the addresses 10 and 11 are not used. Access L ED 8 thus initiates the system to load the value DW#16#00000000 to the accumulator.

10.1.4 Recommendations for Setting the CPU Parameters

CPU parameters that determine cyclic behavior

You specify the CPU parameters that determine the cyclic behavior of the system on the "Cycle/Clock memory" tab.

Recommended settings:

- As long a scan cycle monitoring time as possible (e.g.6000 ms)

- OB 85 call when there is an I/O access error: only with incoming and outgoing errors

Number of messages in the diagnostics buffer

You specify the number of messages in the diagnostics buffer on the "Diagnostics/Clock" tab.

We recommend that you set a large number (1500, for example).

Monitoring time for transferring parameters to modules

You specify this monitoring time on the "Startup" tab. It depends on the configuration of the redundant station. If the monitoring time is too short, the CPU enters the W#16#6547 event in the diagnostics buffer.

For some slaves (e.g. IM 157) these parameters are packed in system data blocks. The transmission time of the parameters depends on the following factors:

- Baud rate of the bus system (high baud rate => short transmission time)

- Size of the parameters and the system data blocks (long parameter => long transmission time)

- Load on the bus system (many slaves => long transmission time);
  Note: The bus load is at its peak during restart of the DP master, for example, following Power OFF/ON

Suggested setting: 600 corresponds to 60 sec.

Note

The special H CPU parameters and the associated monitoring times are calculated automatically. This involves setting a default value for the total memory load of all data blocks specifically for a CPU. If your H system does not link up, check the memory load setting (HW Config -> CPU Properties -> H Parameters -> Work memory used for all data blocks).

Notice

A CP443-5 Extended may only be used with transmission rates of 1.5 Mbps in an S7-400H or S7-400F system with interconnected DP/PA or Y-Link (IM157, order no. 6GK7443-5DX03.) Help: see FAQ 11168943 under http://www.siemens.com/automation/service&support.

10.1.5 Configuring Networks

The redundant S7 connection is a separate connection type of the "Configure Networks" application. The following communication peers can communicate with each other:

- S7-400 redundant station   –>   S7-400 redundant station
  (with 2 redundant CPUs)   (with 2 redundant CPUs)

- S7-400 station (with 1 redundant CPU)   –>   redundant S7-400 station
                                                    (with 2 redundant CPUs)

- S7-400 station (with 1 redundant CPU)   –>   S7-400 station
                                                    (with 1 redundant CPU)

- SIMATIC PC stations   –>   redundant S7-400 station
                                                    (with 2 redundant CPUs)

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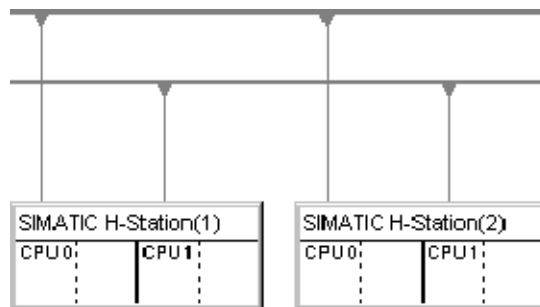
- If only one DP master system is available – in practice typically fiber-optic cables – four connection paths are used for a connection between two redundant stations. All the CPs are in this subnet:
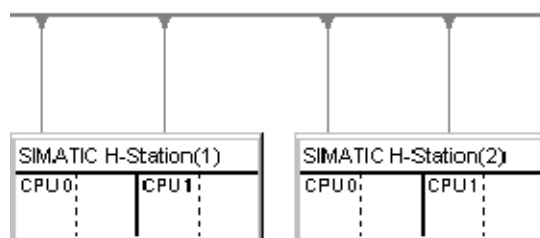
Downloading the network configuration to the redundant station

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

10.2 Programming Device Functions in STEP 7

Display in SIMATIC Manager

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

- In the offline view, the S7 program appears only under CPU0 of the redundant station. No S7 program is visible under CPU1.

- In the online view, the S7 program appears under both CPUs and can be selected in both locations.

Communication functions

For PG communication functions such as 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 coupling – in other words, in stand–alone mode – initially affect only the edited CPU. The blocks are applied by the master CPU to the standby CPU during the next coupling 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 **11**

One factor that is crucial to the uninterrupted operation of the redundant PLC is the replacement of failed components in run. Quick repairs will recover redundancy.

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 Programmable Controllers, Hardware and Installation*

In Section	Description	On Page
11.1	Failure and replacement of components in central racks and expansion racks	11-2
11.2	Failure and replacement of components of the distributed I/O	11-12

11.1 Failure and replacement of components in central racks and expansion racks

Which components can be replaced?

The following components can be replaced during operation:

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

11.1.1 Failure and replacement of a CPU (redundant CPU)

Complete replacement of the CPU is not always necessary. If only load memory fails, it suffices to replace the corresponding memory module. Both cases are described below.

Starting situation for replacement of the complete CPU

Failure	System reaction
The S7-400H is in redundant mode and a **CPU** fails.	• Partner CPU switches to stand–alone mode. • Partner CPU reports the event in the diagnostics buffer and in OB 72.

Requirements for a replacement

The module replacement described below is possible only if the "new" CPU

- has the same operating system version as the failed CPU and

- and if it is equipped with the same memory as the failed CPU.

Notice

New CPUs are always delivered with the latest operating system version. To be able to use this type of CPU as a replacement module, you must create an operating system update card for the operating system version of the failed CPU and use it to transfer the operating system to the replacement CPU.

Procedure

To change a CPU:

Step	What Has To Be Done?	How Does the System React?
1	Turn off the power supply module.	• The entire subsystem is switched off (system operating in stand–alone mode).
2	Replace the central processing unit.Make sure the rack number is set correctly.	-
3	Insert in the synchronization modules.	-
4	Plug in the fiber-optic cable connections of the synchronization modules.	-
5	Switch the power supply module on again.	• CPU executes the self-tests and goes to STOP.

Step	What Has To Be Done?	How Does the System React?
6	Perform a CPU memory reset on the replaced CPU.	-
7	Start the replaced CPU (e.g. STOP→RUN or Start using the PG).	• CPU performs an automatic COUPLING and UPDATE operation. • 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 redundant state and an **access error to the load memory** is executed.	• The relevant CPU goes into STOP and requests a memory reset. • The partner CPU changes over to stand–alone mode.

Procedure

To replace the load memory:

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

11.1.2 Failure and Replacement of a Power Supply Module

Initial situation

Both CPUs are in RUN.

Failure	How Does the System React?
The S7-400H is in redundant mode and one power supply module fails.	• The partner CPU changes over to stand–alone mode. • The partner CPU reports the event in the diagnostics buffer and in OB 72.

Procedure

To change a power supply module in the central rack:

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).	• The entire subsystem is switched off (system operating in stand–alone mode).
2	Replace the module.	-
3	Switch the power supply module on again.	• CPU executes the self-tests. • The CPU automatically performs a COUPLING and UPDATE operation. • The CPU changes to RUN (redundant state) and now operates as the standby CPU.

Note

If you use a redundant power supply (PS 407 10A R), two power supply modules are assigned to one redundant CPU. 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.

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

Initial situation

Failure	How Does the System React?
The S7-400H is in redundant state and an input/output or function module fails.	• Both CPUs report the event in the diagnostics 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 diagnostics 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 diagnostics interrupts are released via the configuration.

11.1.4   Failure and Replacement of a Communication Processor

This section describes the failure and replacement of communication modules for PROFIBUS and Industrial Ethernet.

The failure and replacement of communication processors for the PROFIBUS DP are described in Section 11.2.1

Initial situation

Failure	How Does the System React?
The S7-400H is in redundant state and one communication processor fails.	• Both CPUs report the event in the diagnostics 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 parameter 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 configured by the appropriate CPU.
3	Turn the module back on.	The module resumes communications (system establishes communication connection automatically).

11.1.5 Failure and replacement of a synchronization module or fiber-optic cable

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

- Failure of a synchronization module or fiber-optic cable
- Successive failure of the two synchronization modules or fiber-optic cables
- Simultaneous failure of the two synchronization modules or fiber-optic cables

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 module) have been replaced, LEDs IFM1F and IFM2F go out.

Initial situation

Failure	How Does the System React?
Failure of a fiber–optic cable or synchronization module: The S7-400H is in redundant state and a fiber-optic cable or a synchronization module fails.	• Master CPU reports the event in the diagnostics buffer and via OB 72. • Master CPU remains in RUN mode; standby CPU goes into STOP • The diagnostics LED of the synchronization module is lit

Procedure

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

Step	What Has To Be Done?	How Does the System React?
1	First, check the optical fiber cable.[1]	-
2	Start the standby CPU (e.g. STOP→RUN or Start via programming device).	The following reactions are possible: 1. CPU goes into RUN mode. 2. CPU goes into STOP mode. In this case continue with step 3.
3	Unplug the faulty synchronization module from the standby CPU.	-
4	Plug the new synchronization module into the standby CPU.	-
5	Plug in the fiber-optic cable connections of the synchronization modules.	• The diagnostics LED of the synchronization module is now switched off • Both CPUs report the event in the diagnostics buffer
6	Start the standby CPU (e.g. STOP→RUN or Start via programming device).	The following reactions are possible: 1. CPU goes into RUN mode. 2. CPU goes into 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 module from the master CPU.	• Master CPU executes insert/remove-module interrupt OB 83 and redundancy error OB 72 (incoming).
8	Plug the new synchronization module 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 modules.	-
10	Start the standby CPU (e.g. STOP→RUN or Start via programming device).	• CPU performs automatic LINK-UP and UPDATE. • CPU changes to RUN (redundant state) and now operates as the standby CPU.

Note

If both fiber-optic cables or synchronization modules 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?
Failure of both fiber–optic cables or of the synchronization module: The S7-400H is in redundant state and **both** fiber-optic cables or synchronization modules fail.	• Both CPUs report the event in the diagnostics buffer and via OB 72. • Both CPUs become the master CPU and remain in RUN mode. • The diagnostics LED of the synchronization module is lit

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 (e.g. STOP→RUN or Start via programming device).	• CPU performs automatic LINK-UP and UPDATE. • CPU changes to RUN (redundant state) and now operates as the standby CPU.

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

The IM 460 and IM 461 interface modules provide functions for connecting expansion modules.

Initial situation

Failure	How Does the System React?
The S7-400H is in redundant state and one interface module fails.	• Connected expansion unit is turned off. • Both CPUs report the event in the diagnostics 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 stand–alone 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.

11.2 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

Replacement of I/O and function modules located in remote stations is described in Section 11.1.3.

11.2.1 Failure and Replacement of a PROFIBUS-DP Master

Initial situation

Failure	How Does the System React?

The S7-400H is in redundant state 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 redundant system changes over stand–alone 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.

11.2.2   Failure and Replacement of a Redundant PROFIBUS-DP Interface Module

Initial situation

Failure	How Does the System React?
The S7-400H is in redundant state and a PROFIBUS-DP interface module (IM 153-2, IM 157) fails.	Both CPUs report the event in the diagnostics 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 (outgoing event). • Redundant access to the station is again possible for the system.

11.2.3 Failure and Replacement of a PROFIBUS-DP Slave

Initial situation

Failure	How Does the System React?
The S7-400H is in redundant state and one DP slave fails.	Both CPUs report the event in the diagnostics 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 (outgoing event). • DP slave can be addressed by the relevant DP master system.

11.2.4 Failure and Replacement of PROFIBUS-DP Cables

Initial situation

Failure	How Does the System React?
The S7-400H is in redundant state 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 (outgoing event) DP slaves can be addressed via the DP master system. • With switched I/O: I/O redundancy error OB70 (outgoing event). DP slaves can be addressed via both DP master systems.

Modifications to the System During Operation 12

In addition to the options of hot–swapping of failed modules as described in chapter 11, both the 417-4H and the 414-4H CPUs allow system modifications without interruption of the active program.

The procedure depends on whether you are working on your user software in PCS 7 or STEP 7.

The starting point in the structure of each one of the procedures described below is redundant state, with the target set on this mode (see chapter 6.2).

Notice

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 redundant system can result in its availability being restricted or even failure of the entire programmable logic controller.

Security-relevant components are not taken into account in this description. For more details of dealing with fail-safe systems refer to the manual *S7-400F and S7-400FH Programmable Controllers*.

12.1 Possible Hardware Modifications

How is a hardware change made?

If the hardware components concerned are suitable for unplugging or plugging in live the hardware modification can be carried out in redundant state. However, the redundant redundant system must be operated temporarily in stand–alone mode, because any download of new hardware configuration data in redundant mode would inevitably lead its STOP. The process is then controlled only by one CPU, while you can carry out the relevant changes at the partner CPU.

Note

During a hardware change, you can either remove or add modules. If you want to alter your H system in such a that you remove some modules and add others, you have to make two hardware changes.

Notice

Always download delta configuration to the CPU using the "Configure hardware" function.

Load memory data of the redundant CPUs must be updated several times in the process. It is therefore advisable to expand the integrated load memory with a RAM module, at least temporarily.

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 you can not obtain a RAM module with a capacity to match FLASH memory space, split the relevant actions in your configuration and program modifications into several smaller steps, in order to provide sufficient space in the integrated load memory.

⚠ **Caution**

After you have modified any HW configurations, always reconnect the synchronization links between the redundant CPUs **before** you restart or power up the standby CPU. After the power supplies of the CPUs are switched on, the IFM1F and IFM2F LEDs which are used to indicate faults on the interfaces for memory modules **must be switched off 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-sided I/O module).

Notice

Always switch of power before you install or remove the IM460 and IM461 interface modules, external CP443-5 Extended DP master interface module and the corresponding connecting cables.

- Adding or removing components of the remote input/output station, such as
    - DP slaves with a redundant interface module (e.g. ET 200M, DP/PA link or Y link)
    - One-sided DP slaves (in any DP master system)
    - Modules in modular DP slaves
    - DP/PA couplers
    - PA devices
- Changing certain CPU parameters
- Changing the Memory Components of the CPU
- Reconfiguration of a module
- Assign module to another process image partition
- Upgrading the CPU version
- Change of master with only one more available redundancy connection.

With all modifications observe the rules for the configuration of a redundant station (see Section 2.1).

Refer to the information text in the "Hardware Catalog" window to determine which ET 200M modules (signal modules and function modules) can be reconfigured during ongoing operation. The special reactions of individual modules are described in the respective technical documentation.

What should I note at the system planning stage?

In order for switched I/O to be able to be expanded during 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 million bps). 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), DP/PA links and Y links must always be installed with an active backplane bus and fitted with all the bus modules required, wherever possible, because the bus modules can not be installed and removed during operation.

- Always terminate both ends of PROFIBUS DP and PROFIBUS PA bus cables using active bus terminators in order to ensure proper termination of the cables while you are reconfiguring the system.

- 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 deleted and created again. In other words, SFC 22 (CREATE_DB) and SFC 23 (DEL_DB) may not be applied to DB numbers occupied by loaded DBs.

- Always ensure that the current status of the user program is available as STEP 7 project in a block format at the PG/ES when you modify the system configuration. It is not satisfactory to upload the user program from one of the CPUs to the PG/ES, or to compile the code again from an STL source.

Modification of the hardware configuration

With a few exceptions, all elements of the configuration can be modified in run. Usually, any configuration changes will also affect 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 redundant state. The procedure depends on the software used. For more details refer to the *Programming with STEP 7 V5.1* manual and the *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.

- In the case of IM 153-2 bus modules can only be plugged in is the power supply is interrupted.

- Before making a change, check the fault tolerance parameter in HW Config. If this parameter is set to 0, have the parameter calculated again in HW Config using the Properties CPU –> H Parameter.

Notice

The following should be taken into consideration when using redundant I/O modules that you have installed as one–sided modules on the user level (see Chapter 8.5):

Due to the coupling and update process carried out after a system modification, the I/O data of the previous master CPU may be temporarily deleted from the process image until all I/O (delta) data of the ”new” master CPU are written to the process image.

During the initial update of the process image after a system modification, the status might be misinterpreted as total failure of the redundant I/O, or lead to the wrongful assumption of the redundant existence of this I/O. Proper evaluation of the redundancy status is thus not possible until the process image has been fully updated.

This phenomenon does not occur with modules that have been released for redundant operation (see Chapter 8.4).

Preparations

To minimize the time during which the redundant system has to run in stand–alone mode, you should perform the following steps **before** making the hardware change:

- Check whether the CPUs provide sufficient memory capacity for the new configuration data and user program. If necessary, first expand the memory components (see Section 12.7).

- Always ensure that plugged modules which are not configured yet do not have any unwanted influence on the process.

12.2   Adding Components in PCS 7

Initial situation

You have verified that all CPU parameters such as monitoring times match the new program design. If negative, edit the CPU parameters accordingly (see Section 12.6).

The redundant system is operating in redundant mode.

Procedure

Carry out the steps listed below to add hardware components to a redundant system in PCS 7. Details of each step are listed in a subsection.

Step	What Has To Be Done?	Refer to Section
1	Modification of Hardware	12.2.1
2	Offline Modification of the Hardware Configuration	12.2.2
3	Stopping the Standby CPU	12.2.3
4	Loading New Hardware Configuration in the Standby CPU	12.2.4
5	Switch to CPU with modified configuration	12.2.5
6	Transition to redundant state	12.2.6
7	Changing and Loading User Program	12.2.7

Exceptions

This procedure for system modification does not apply in the following cases:

• Use of Free Channels on an Existing Module

• When adding interface modules (see Section 12.2.8)

Note

As of STEP 7 V5.3 SP2, after changing the hardware configuration, you can let the load operation run automatically. This means that you will not have to perform the action steps described in sections 12.2.3 to 12.2.6 anymore. The described system behaviour remains unchanged.

You can find more information in the online help HW Config **Download in module -> Download the station configuration in operating state RUN.**

12.2.1 PCS 7, Step 1: Modification of Hardware

Initial situation

The redundant system is operating in redundant 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.

Notice

With switched I/O: Complete all changes on **one** line of the redundant DP master system first before making changes 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 redundant system continues operation in redundant state.

New components are not yet addressed.

12.2.2   PCS 7, Step 2: Offline Modification of the Hardware Configuration

Initial situation

The redundant system is operating in redundant mode.

Procedure

1. Perform all the modifications to the hardware configuration relating to the added hardware offline. Assign appropriate icons to the new channels to be used.

2. Compile the new hardware configuration, but do **not** load it into the PLC just yet.

Result

The modified hardware configuration is in the PG/ES. The PLC continues to work with the old configuration in redundant state.

Configuring connections

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

12.2.3   PCS 7, Step 3: Stopping the Standby CPU

Initial situation

The redundant system is operating in redundant mode.

Procedure

1. In SIMATIC Manager, select a CPU of the redundant system, and choose the **PLC > Operating Mode** menu command.

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 redundant system works in stand–alone mode. One-sided I/O of the standby CPU are no longer addressed.

Whilst I/O access errors of the one-sided I/O will result in OB85 being called, due to the superior CPU redundancy loss (OB72) these will not be reported. OB 70 (I/O redundancy loss) will not be called.

12.2.4 PCS 7, Step 4: Loading New Hardware Configuration in the Standby CPU

Initial situation

The redundant system is operating in stand–alone mode.

Procedure

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

Notice

The user program and the connection configuration must not be overloaded in stand–alone mode.

Result

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

12.2.5   PCS 7, 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 redundant system, and choose the **PLC > Operating Mode** menu command.

2. In the **Operating Mode** dialog box, click the **Toggle button.**

3. In the **Toggle** dialog box, select the option **with modified configuration** and click the **Toggle** button.

4. Confirm the query that follows with **OK**.

Result

The standby CPU links up, is updated (see Chapter 7) and becomes the master. The former master CPU switches to STOP mode, the redundant system operates with the new hardware configuration in stand–alone mode.

Behavior of the I/O

Type of I/O	One-sided I/O of previous master CPU	One-sided I/O of new master CPU	Switched I/O
Added I/O modules	Are not addressed by the CPU.	Are configured and updated by the CPU. Driver blocks are not yet present. Although process or diagnostics interrupts will be detected, these are not going to be 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 reconfigured [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) The central modules will first be initialized with reset. Output modules briefly output 0 during this time (instead of the configured substitution or hold values).

Reaction to monitoring timeout

The update will be aborted and no change of master takes place if one of the monitored times exceeds the configured maximum. The redundant system remains in stand–alone mode with the previous master CPU and in certain conditions attempts to perform the change of master later. For further information, refer to chapter 7.4.

12.2.6 PCS 7, Step 6: Transition to redundant state

Initial situation

The redundant system works with the new hardware configuration in stand–alone mode.

Procedure

1. In SIMATIC Manager, select a CPU of the redundant system, and choose the **PLC > Operating Mode** menu command.

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 redundant system operates with the new hardware configuration in redundant state.

Behavior of the I/O

Type of I/O	One-sided I/O of standby CPU	One-sided I/O of master CPU	Switched I/O
Added I/O modules	Are configured 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 diagnostics interrupts occurring will be recognized but not reported.	
I/O modules that continue to be present	Are reconfigured [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) The central modules will first be initialized with reset. Output modules briefly output 0 during this time (instead of the configured substitution or hold values).

Reaction to monitoring timeout

The update will be aborted if one of the monitored times exceeds the configured maximum. The redundant system remains in stand–alone mode with the previous master CPU and in certain conditions attempts to perform the link-up and update later. For further information, refer to chapter 7.4.

12.2.7   PCS 7, Step 7: Changing and Loading User Program

Initial situation

The redundant system operates with the new hardware configuration in redundant state.

⚠ **Caution**

The following program modifications are not possible in redundant state 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 parameter settings

2. Assign parameters to the added channel drivers and connect these to the newly assigned icons (see Section 12.2.2).

3. In SIMATIC Manager, select the charts folder and choose the **Extras > Charts > Generate module drivers** menu command.

4. Compile only the modifications in the charts and load these into the PLC.

Notice

Until the first FC is called the value of its coil 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 redundant system processes the entire system hardware with the new user program in redundant state.

12.2.8 Adding Interface Modules in PCS 7

Always switch off power before you install the IM460 and IM461 interface modules, external CP443-5 Extended DP master interface module and the corresponding connecting cables.

Always switch off power to the entire unit. In order to ensure that this does not influence the active process, always set the unit to STOP before you do so.

Procedure

1. Change the hardware configuration offline (see Section 12.2.2)

2. Stop the standby CPU (see Section 12.2.3)

3. Download the new hardware configuration to the standby CPU (see Section 12.2.4)

4. If you want to add to the subsystem of the present standby CPU, carry out the following steps:

   – Switch off power to the standby unit.

   – Plug the new IM460 in the central controller and establish the link to a new expansion unit.

     or

   – Add a new expansion unit to an existing line.

     or

   – Plug in the new external DP master interface, and install a new DP master system.

   – Switch on power to the standby unit.

5. Switch to the CPU which contains the modified configuration (see Section 12.2.5)

6. If you want to expand the subsystem of the original master CPU (currently in STOP mode), carry out the following steps:

   – Switch off power to the standby unit.

   – Plug the new IM460 in the central controller and establish the link to a new expansion unit.

     or

   – Add a new expansion unit to an existing line.

     or

   – Plug in the new external DP master interface, and install a new DP master system.

   – Switch on power to the standby unit.

7. Switch to redundant state (see Section 12.2.6)

8. Modify and download the user program (see Section 12.2.7)

12.3 Removing Components in PCS 7

Initial situation

You have verified that all CPU parameters such as monitoring times match the new program design. If necessary you must first change the CPU parameters accordingly (see Section 12.6).

The modules to be removed and their connected sensors and actuators are no longer of any significance to the process to be controlled. The redundant system is operating in redundant mode.

Procedure

Carry out the steps listed below to remove hardware components from a redundant system in PCS 7. 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	12.3.1
II	Changing and Loading User Program	12.3.2
III	Stopping the Standby CPU	12.3.3
IV	Loading New Hardware Configuration in the Standby CPU	12.3.4
V	Switch to CPU with modified configuration	12.3.5
VI	Transition to redundant state	12.3.6
VII	Modification of Hardware	12.3.7

Exceptions

This procedure for system modification should not be used to remove interface modules (see Section 12.3.8).

Note

As of STEP 7 V5.3 SP2, after changing the hardware configuration, you can let the load operation run automatically. This means that you will not have to perform the action steps described in sections 12.3.3 to 12.3.6 anymore. The described system behaviour remains unchanged.

You can find more information in the online help HW Config **Download in module -> Download the station configuration in operating state RUN.**

12.3.1   PCS 7, Step I: Offline Modification of the Hardware Configuration

Initial situation

The redundant system is operating in redundant mode.

Procedure

1. Perform offline only the configuration modifications relating to the hardware to be removed. As you do, delete the icons to the channels that are no longer used.

2. Compile the new hardware configuration, but do **not** load it into the PLC just yet.

Result

The modified hardware configuration is in the PG/ES. The PLC continues to work with the old configuration in redundant state.

12.3.2   PCS 7, Step II: Changing and Loading User Program

Initial situation

The redundant system is operating in redundant mode.

⚠ **Caution**

The following program modifications are not possible in redundant state 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 parameter settings

2. In SIMATIC Manager, select the charts folder and choose the **Extras > Charts > Generate module drivers** menu command.

   This removes the driver blocks that are no longer required.

3. Compile only the modifications in the charts and load these into the PLC.

Notice

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

Result

The redundant system continues operation in redundant state. The modified user program will no longer attempt to access the hardware to be removed.

12.3.3 PCS 7, Step III: Stopping the Standby CPU

Initial situation

The redundant system is operating in redundant 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 redundant system, and choose the **PLC > Operating Mode** menu command.

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 redundant system works in stand–alone mode. One-sided I/O of the standby CPU are no longer addressed.

12.3.4 PCS 7, Step IV: Loading New Hardware Configuration in the Standby CPU

Initial situation

The redundant system is working in stand–alone mode.

Procedure

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

Notice

The user program and the connection configuration must not be overloaded in stand–alone mode.

Result

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

12.3.5 PCS 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 redundant system, and choose the **PLC > Operating Mode** menu command.

2. In the **Operating Mode** dialog box, click the **Toggle button.**

3. In the **Toggle** dialog box, select the option **with modified configuration** and click the **Toggle** button.

4. Confirm the query that follows with **OK**.

Result

The standby CPU links up, is updated (see Chapter 7) and becomes the master. The former master CPU switches to STOP mode, the redundant system works with the new hardware configuration in stand–alone mode.

Behavior of the I/O

Type of I/O	One-sided I/O of previous master CPU	One-sided 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 reconfigured [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) The central modules will first be initialized with reset. Output modules briefly output 0 during this time (instead of the configured substitution or hold values).

Reaction to monitoring timeout

The update will be aborted and no change of master takes place if one of the monitored times exceeds the configured maximum. The redundant system remains in stand–alone mode with the previous master CPU and in certain conditions attempts to perform the change of master later. For further information, refer to chapter 7.4.

12.3.6   PCS 7, Step VI: Transition to redundant state

Initial situation

The redundant system works with the new hardware configuration in stand–alone mode.

Procedure

1. In SIMATIC Manager, select a CPU of the redundant system, and choose the **PLC > Operating Mode** menu command.

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 redundant system operates with the new hardware configuration in redundant state.

Behavior of the I/O

Type of I/O	One-sided I/O of standby CPU	One-sided 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 reconfigured [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) The central modules will first be initialized with reset. Output modules briefly output 0 during this time (instead of the substitution or hold values configured).

Reaction to monitoring timeout

The update will be aborted if one of the monitored times exceeds the configured maximum. The redundant system remains in stand–alone mode with the previous master CPU and in certain conditions attempts to perform the link-up and update later. For further information, refer to chapter 7.4.

12.3.7 PCS 7, Step VII: Modification of hardware

Initial situation

The redundant system operates with the new hardware configuration in redundant state.

Procedure

1. Disconnect all the sensors and actuators from the components to be removed.

2. Unplug modules of the one-sided 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.

Notice

With switched I/O: Complete all changes on **one** line of the redundant DP master system first before making changes 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 redundant system continues to work in redundant state.

12.3.8   Removing Interface Modules in PCS 7

Always switch off power before you install the IM460 and IM461 interface modules, external CP443-5 Extended DP master interface module and the corresponding connecting cables.

Always switch off power to the entire unit. In order to ensure that this does not influence the active process, always set the unit to STOP before you do so.

Procedure

1. Change the hardware configuration offline (see Section 12.3.1)

2. Modify and download the user program (see Section 12.3.2)

3. Stop the standby CPU (see Section 12.3.3)

4. Download the new hardware configuration to the standby CPU (see Section 12.3.4)

5. If you want to remove an interface module from the subsystem of the current standby CPU, carry out the following steps:

    – Switch off power to the standby unit.

    – Remove an IM460 from the central controller.

      or

    – Remove an expansion unit from an existing line.

      or

    – Remove an external DP master interface.

    – Switch on power to the standby unit.

6. Switch to the CPU which contains the modified configuration (see Section 12.3.5)

7. If you want to remove an interface module from the subsystem of the original master CPU (currently in STOP mode), carry out the following steps:

    – Switch off power to the standby unit.

    – Remove an IM460 from the central controller.

      or

    – Remove an expansion unit from an existing line.

      or

    – Remove an external DP master interface.

    – Switch on power to the standby unit.

8. Switch to redundant state (see Section 12.3.6)

12.4 Adding Components in STEP 7

Initial situation

You have verified that all CPU parameters such as monitoring times match the new program design. If necessary you must first change the CPU parameters accordingly (see Section 12.6).

The redundant system is operating in redundant mode.

Procedure

In order to add hardware components to a redundant 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	12.4.1
2	Offline Modification of the Hardware Configuration	12.4.2
3	Expanding and Loading Organization Blocks	12.4.3
4	Stopping the Standby CPU	12.4.4
5	Loading New Hardware Configuration in the Standby CPU	12.4.5
6	Switch to CPU with modified configuration	12.4.6
7	Transition to redundant state	12.4.7
8	Changing and Loading User Program	12.4.8

Exceptions

This procedure for system modification does not apply in the following cases:

- Use of Free Channels on an Existing Module

- When adding interface modules (see Section 12.4.9)

Note

As of STEP 7 V5.3 SP2, after changing the hardware configuration, you can let the load operation run automatically. This means that you will not have to perform the action steps described in sections 12.4.4 to 12.4.7 anymore. The described system behaviour remains unchanged.

You can find more information in the online help HW Config **Download in module -> Download the station configuration in operating state RUN.**

12.4.1   STEP 7, Step 1: Adding the hardware

Initial situation

The redundant system is operating in redundant 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.

Notice

With switched I/O: Complete all changes on **one** line of the redundant DP master system first before making changes to the second line.

---

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

Result

The insertion of non–configured modules will have no effect on the user program. The same applies to adding DP stations.

The redundant system continues operation in redundant mode.

New components are not yet addressed.

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

Initial situation

The redundant system is operating in redundant 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** download it to the PLC yet.

Result

The new hardware configuration is available on the PG. The PLC continues operation with the old configuration in redundant mode.

Configuring connections

The interconnections with added CPs must be configured on both connection partners **after** you completed the HW modification.

12.4.3 STEP 7, Step 3: Expanding and downloading OBs

Initial situation

The redundant system is operating in redundant mode.

Procedure

1. Verify that the interrupt OBs 4x, 82, 83, 85, 86, OB88 and 122 react to any alarms of the new components as intended.

2. Download the modified OBs and the corresponding program elements to the PLC.

Result

The redundant system is operating in redundant mode.

12.4.4   STEP 7, Step 4: Stopping the standby CPU

Initial situation

The redundant system is operating in redundant mode.

Procedure

1. In SIMATIC Manager, select a CPU of the redundant system, then select the **PLC > Operating Mode** command.
2. From the **Operating Mode** dialog box, select the standby CPU, then click **Stop**.

Result

The standby CPU goes into STOP, the master CPU remains in RUN. The redundant system now operates in stand–alone mode. One-sided I/O of the standby CPU are no longer addressed. OB 70 (I/O redundancy loss) is not called due to the higher–priority CPU redundancy loss (OB72).

12.4.5   STEP 7, Step 5: Downloading the new HW configuration to the standby CPU

Initial situation

The redundant system is operating in stand–alone mode.

Procedure

Download the compiled hardware configuration to the standby CPU while it is in STOP.

Notice

The user program and the connection configuration may not be downloaded in stand–alone mode.

Result

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

12.4.6   STEP 7, Step 6: Switching to the CPU which contains the modified data

Initial situation

The modified hardware configuration was downloaded to the standby CPU.

Procedure

1. In SIMATIC Manager, select a CPU of the redundant system, then select the **PLC > Operating Mode** command.

2. On the **Operating Mode** dialog box, click **Toggle...**

3. From the **Toggle** dialog box, set the **with modified configuration** check box, then click **Toggle**.

4. Confirm the safety prompt with **OK**.

Result

The standby CPU will be coupled and updated, and assumes master mode. The previous master CPU goes into STOP, the redundant system now operates with the new HW configuration in stand–alone mode.

Reaction of the I/O

Type of I/O	One-sided I/O of previous master CPU	One-sided I/O of new master CPU	Switched I/O
Added I/O modules	Are not addressed by the CPU.	Are configured and updated by the CPU. Output modules briefly output the configured substitution values.	
I/O modules that continue to be present	Are no longer addressed by the CPU. Output modules output the configured substitution or hold values.	Are reconfigured [1] and updated by the CPU.	Continue operation without interruption.
Added DP stations	Are not addressed by the CPU.	as for added I/O modules (see above)	

1) The central modules will be initialized with reset. Output modules briefly output 0 during this time (instead of the configured substitution or hold values).

Reaction to monitoring timeout

If one of the monitored times exceeds the configured maximum, the update will be interrupted and no change of master takes place. The redundant system remains in stand–alone mode with the previous master CPU, and under certain conditions retries to perform the change of masters. For detailed information, refer to Section 7.4.

12.4.7 STEP 7, Step 7: System transition to redundant mode

Initial situation

The redundant system operates with the new HW configuration in stand–alone mode.

Procedure

1. In SIMATIC Manager, select a CPU of the redundant system, then select the **PLC > Operating Mode** command.

2. From the **Operating Mode** dialog box, select the standby CPU, then click **Restart (warm restart)**.

Result

The standby CPU will be coupled and updated. The redundant system is now operating with the new hardware configuration in redundant mode.

Reaction of the I/O

Type of I/O	One-sided I/O of standby CPU	One-sided I/O of master CPU	Switched I/O
Added I/O modules	Are configured and updated by the CPU. The output modules temporarily output the configured substitution values.	Are updated by the CPU.	Are updated by the CPU. Generate insertion interrupt; must be ignored in OB83.
I/O modules that continue to be present	Are reconfigured [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) The central modules will be initialized with reset. Output modules temporarily output 0 during this time, (instead of the configured substitution or hold values.

Reaction to monitoring timeout

If one of the monitored times exceeds the set maximum the update will be interrupted. The redundant system remains in stand–alone mode with the previous master CPU, and under certain conditions retries to couple and update later. For detailed information, refer to chapter 7.4.

12.4.8   STEP 7, Step 8: Editing and downloading the user program

Initial situation

The redundant system operates with the new hardware configuration in redundant mode.

Restrictions

⚠ **Caution**

Any attempts to modify the structure of an FB interface or the instance data of an FB in redundant mode will lead to a system STOP at both CPUs.

Procedure

1. Adapt the program to the new hardware configuration.

   You can add, edit or remove OBs, FBs, FCs and DBs.

2. Download only the delta data to the PLC.

3. Configure the interconnections for the new CPs on both communication partners and download the delta data to the PLC.

Result

The redundant system processes the entire system hardware in redundant mode, based on the new user program.

12.4.9   Adding Interface Modules in STEP 7

Always switch off power before you install the IM460 and IM461 interface modules, external CP443-5 Extended DP master interface module and the corresponding connecting cables.

The corresponding unit must be isolated from the power supply. Influences on the process can only be avoided by these actions if the unit is in STOP mode.

Procedure

1. Edit the hardware configuration offline (see chapter 12.4.2)

2. Expand and download the organization blocks (see chapter 12.4.3)

3. Stop the standby CPU (see chapter 12.4.4)

4. Download the new hardware configuration to the standby CPU (see chapter 12.4.5)

5. To expand the unit of the present standby CPU:

   – Turn off the power supply to the standby unit.

   – Plug the new IM460 into the central rack, then establish the link to a new expansion unit.

      or

   – Add a new expansion unit to an existing segment.

      or

   – Plug in the new external DP master interface, and install a new DP master system.

   – Turn on the power supply for the standby unit again.

6. Switch to the CPU which contains the modified configuration (see chapter 12.4.6)

7. To expand the unit of the original master CPU (currently in STOP mode):

   – Turn off the power supply for the standby unit.

   – Insert the new IM460 into the central rack, then establish the link to a new expansion unit.

      or

   – Add a new expansion unit to an existing segment.

      or

   – Insert the new external DP master interface, and install a new DP master system.

   – Turn on the power supply for the standby unit again.

8. System transition to redundant mode (see chapter 12.4.7)

9. Edit and download the user program (see chapter 12.4.8)

12.5 Removing components in STEP 7

Initial situation

You have verified that all CPU parameters such as monitoring times match the new program design. If negative, edit the CPU parameters accordingly (see chapter 12.6).

The modules to be removed and their connected sensors and actuators are no longer of any significance to the process to be controlled. The redundant system is operating in redundant mode.

Procedure

Carry out the steps listed below to remove hardware components from a redundant system in STEP 7. Details of each step are listed in a subsection.

Step	What Has To Be Done?	Refer to Section
I	Editing the hardware configuration offline	12.5.1
II	Editing and downloading the user program	12.5.2
III	Stopping the standby CPU	12.5.3
IV	Downloading the new hardware Configuration to the standby CPU	12.5.4
V	Switching to the CPU which contains the modified configuration	12.5.5
VI	System status transition to redundant mode	12.5.6
VII	Modifying hardware	12.5.7
VIII	Modifying and Loading Organization Blocks	12.5.8

Exceptions

This overall system modification procedure does not apply to the removal of interface modules (see chapter 12.5.9).

Note

As of STEP 7 V5.3 SP2, after changing the hardware configuration, you can let the load operation run automatically. This means that you will not have to perform the action steps described in sections 12.5.3 to 12.5.6 anymore. The described system behaviour remains unchanged.

You can find more information in the online help HW Config **Download in module -> Download the station configuration in operating state RUN.**

12.5.1   STEP 7, Step I: Editing the hardware configuration offline

Initial situation

The redundant system is operating in redundant mode.

Procedure

1. Edit the hardware configuration offline according to the hardware to be removed.

2. Compile the new hardware configuration, but do **not** download it to the PLC yet.

Result

The modified hardware configuration is available on the PG. The PLC continues operation with the old configuration in redundant mode.

12.5.2   STEP 7, Step II: Editing and downloading the user program

Initial situation

The redundant system is operating in redundant mode.

Restrictions

⚠️ **Caution**

Any attempts to modify the structure of an FB interface or the instance data of an FB in redundant mode will lead to a system STOP at both CPUs.

Procedure

1. Edit only the program elements related to the hardware removal.

   You can add, edit or remove OBs, FBs, FCs and DBs.

2. Download only the delta data to the PLC.

Result

The redundant system continues operation in redundant mode. The new user program will no longer attempt to access the hardware to be removed.

12.5.3 STEP 7, Step III: Stopping the standby CPU

Initial situation

The redundant system is operating in redundant 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 redundant system, then select the **PLC > Operating Mode** command.

2. From the **Operating Mode** dialog box, select the standby CPU, then click **Stop**.

Result

The standby CPU goes into STOP, the master CPU remains in RUN. The redundant system now operates in stand–alone mode. One-sided I/O of the standby CPU are no longer addressed.

12.5.4 STEP 7, Step IV: Downloading the new hardware configuration to the Standby CPU

Initial situation

The redundant system is operating in stand–alone mode.

Procedure

Download the compiled hardware configuration to the standby CPU while it is in STOP mode.

Notice

The user program and the connection configuration may not be downloaded in stand–alone mode.

Result

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

12.5.5 STEP 7, Step V: Switching to the CPU which contains the modified configuration

Initial situation

The modified hardware configuration was downloaded to the standby CPU.

Procedure

1. In SIMATIC Manager, select a CPU of the redundant system, then select the **PLC > Operating Mode** command.

2. On the **Operating Mode** dialog box, click **Toggle.**

3. From the **Toggle** dialog box, set the **with modified configuration** check box, and then click **Toggle**.

4. Confirm the safety prompt with **OK**.

Result

The standby CPU will be coupled and updated (see chapter 7) and assumes master mode. The previous master CPU goes into STOP mode, and the redundant system continues operation in stand–alone mode.

Behavior of the I/O

Type of I/O	One-sided I/O of previous master CPU	One-sided 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 substitution or hold values.	Are reconfigured [2] and updated by the CPU.	Continue operation 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) The central modules will first be initialized with reset. Output modules briefly output 0 during this time (instead of the substitution or hold values configured).

Reaction to monitoring timeout

If one of the monitored times exceeds the configured maximum, the update will be interrupted and no change of master takes place. The redundant system remains in stand–alone mode with the previous master CPU, and under certain conditions retries to couple and update later. For further information, refer to chapter 7.4.

12.5.6   STEP 7, Step VI: System transition to redundant mode

Initial situation

The redundant system operates with the new (restricted) hardware configuration in stand–alone mode.

Procedure

1. In SIMATIC Manager, select a CPU of the redundant system, then select the **PLC > Operating Mode** command.

2. From the **Operating Mode** dialog box, select the standby CPU, and then click **Restart (warm restart)**.

Result

The standby CPU couples and will be updated. The redundant system is operating in redundant mode.

Behavior of the I/O

Type of I/O	One-sided I/O of standby CPU	One-sided 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 reconfigured [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) The central modules will first be initialized with reset. Output modules briefly output 0 during this time (instead of the configured substitution or hold values).

Reaction to monitoring timeout

The update will be aborted if one of the monitored times exceeds the configured maximum. The redundant system remains in stand–alone mode with the previous master CPU, and under certain conditions retries to couple and update later. For further information, refer to chapter 7.4.

12.5.7   STEP 7, Step VII: Modification of hardware

Initial situation

The redundant system operates with the new hardware configuration in redundant state.

Procedure

1. Disconnect all the sensors and actuators from the components you want to remove.

2. Remove the relevant components from the system.

    – Remove the central modules from the rack.

    – Remove the modules from modular DP stations

    – Remove DP stations from DP master systems.

Notice

With switched I/O: always complete all changes on **one** segment of the redundant DP master system before you modify the next segment.

Result

The removal of non–configured modules does not influence the user program. The same applies to the removal of DP stations.

The redundant system continues operation in redundant state.

12.5.8   STEP 7, Step VIII: Editing and downloading organization blocks

Initial situation

The redundant system is operating in redundant mode.

Procedure

1. Make sure that the interrupt OBs 4x and 82 no longer contain any interrupts of the removed components.

2. Download the modified OBs and the corresponding program elements to the PLC.

Result

The redundant system is operating in redundant mode.

12.5.9   Removing interface modules in STEP 7

Always switch off power before you install or remove the IM460 and IM461 interface modules, the external DP master interface module CP443-5 Extended and the associated connecting cables.

Always switch off power to the entire unit. In order to ensure that this does not influence the active process, always set the unit to STOP before you do so.

Procedure

1. Edit the hardware configuration offline (see chapter 12.5.1)

2. Edit and download the user program (see chapter 12.5.2)

3. Stop the standby CPU (see chapter 12.5.3)

4. Download the new hardware configuration to the standby CPU (see chapter 12.5.4)

5. To remove an interface module from the unit of the current standby CPU:

   – Switch off the power to the standby unit.

   – Remove an IM460 from the PLC.

      or

   – Remove an expansion unit from an existing segment.

      or

   – Remove an external DP master interface.

   – Switch on power to the standby unit again.

6. Switch to the CPU which contains the modified configuration (see chapter 12.5.5)

7. To remove an interface module from the unit of the original master CPU (currently in STOP mode):

   – Switch off power to the standby unit.

   – Remove an IM460 from the central controller.

      or

   – Remove an expansion unit from an existing segment.

      or

   – Remove an external DP master interface.

   – Switch on power to the standby unit.

8. Switch the system to redundant mode (see chapter 12.5.6)

9. Edit and download the organization blocks (see chapter 12.5.8)

12.6 Editing CPU parameters

Only certain CPU parameters (object properties) can be edited in run. These are highlighted on the screen forms by blue text. If you have set blue as the color for dialog box text on the Windows Control Panel, the editable parameters are indicated in black characters.

Notice

If you edit any protected parameters, the system will reject any attempt to changeover to the CPU containing those modified parameters. The error event W#16#5966 will be triggered and written to the diagnostics buffer, and you will then have to restore the previous valid values in the parameter configuration.

Table 12-1 Editable CPU parameters

Tab	Editable parameters
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 communication
	Size of the process image of inputs *)
	Size of the process image of outputs *)
Memory	Local data for the various priority classes *)
	Communication resources: maximum number of communication requests .You may only increase the configured value of this parameter. *)
Time-of-day interrupts (for every time-of-day interrupt OB)	"Active" check box
	"Execution" list box
	Starting date
	Time
Watchdog interrupt (for each watchdog interrupt OB)	Execution
	Phase offset
Diagnostics/clock	Correction factor
Security	Security level and password

*) Modifying these parameters also modifies the memory content. Data blocks genrerated through SFC are deleted.

Table 12-1   Editable CPU parameters, continued

Tab	Editable parameters
Fault-tolerant parameters	Test scan cycle time
	maximum scan-cycle time extension
	Maximum communication delay
	Maximum retention time for priority classes > 15
	Minimum I/O hold time

The selected new values should match both the currently loaded and the new user program.

Initial situation

The redundant system is operating in redundant mode.

Procedure

To edit the CPU parameters of a redundant system, follow the steps outlined below. Each step is described in a separate subsection.

Step	What Has To Be Done?	Refer to chapter
A	Editing CPU parameters offline	12.6.1
B	Stopping the standby CPU	12.6.2
C	Downloading modified CPU parameters to the standby CPU	12.6.3
D	Changeover to the CPU which contains the modified configuration	12.6.4
E	System transition to redundant mode	12.6.5

Note

As of STEP 7 V5.3 SP2, after changing the hardware configuration, you can let the load operation run automatically. This means that you will not have to perform the action steps described in sections 12.6.2 to 12.6.5 anymore. The described system behaviour remains unchanged.

You can find more information in the online help HW Config **Download in module -> Download the station configuration in operating state RUN.**

12.6.1 Step A: Editing CPU parameters offline

Initial situation

The redundant system is operating in redundant mode.

Procedure

1. Edit the relevant CPU properties offline in HW Config.

2. Compile the new hardware configuration, but do **not** download it to the PLC yet.

Result

The modified hardware configuration is available on the PG/ES. The PLC continues operation with the old configuration in redundant mode.

12.6.2 Step B: Stopping the standby CPU

Initial situation

The redundant system is operating in redundant mode.

Procedure

1. In SIMATIC Manager, select a CPU of the redundant system, then select the **PLC > Operating Mode** command.

2. From the **Operating Mode** dialog box, select the standby CPU, and then click **Stop**.

Result

The standby CPU goes into STOP, the master CPU remains in RUN mode, the redundant system operates in stand–alone mode. One-sided I/O of the standby CPU are no longer addressed.

12.6.3 Step C: Downloading modified CPU parameters to the standby CPU

Initial situation

The redundant system is operating in stand–alone mode.

Procedure

Download the compiled hardware configuration to the standby CPU while it is in STOP mode.

Notice

The user program and connection configuration may not be downloaded in stand–alone mode.

Result

The modified CPU parameters in the new hardware configuration of the standby CPU do not yet have an effect on the active process.

12.6.4 Step D: Changeover to the CPU which contains the modified configuration

Initial situation

The modified hardware configuration was downloaded to the standby CPU.

Procedure

1. In SIMATIC Manager, select a CPU of the redundant system, then select the **PLC > Operating Mode** command.

2. On the **Operating Mode** dialog box, click **Toggle.**

3. On the **Toggle** dialog box, set the **with modified configuration** check box, and then click **Toggle**.

4. Confirm the safety prompt with **OK**.

Result

The standby CPU will be coupled and updated, and assumes master mode. The previous master CPU goes into STOP, and the redundant system continues operation in stand–alone mode.

Reaction of the I/O

Type of I/O	One-sided I/O of previous master CPU	One-sided I/O of new master CPU	Switched I/O
I/O modules	Are no longer addressed by the CPU. Output modules output the configured substitution or hold values.	Are reconfigured [1] and updated by the CPU.	Continue operation without interruption.

1) in addition, the central modules will be initialized with reset. Output modules briefly output 0 during this time (instead of the configured substitution or hold values).

Reaction to monitoring timeout

The update will be aborted and no change of master takes place if one of the monitored times exceeds the configured maximum. The redundant system remains in stand–alone mode with the previous master CPU, and under certain conditions retries to couple and update later. For further information, refer to chapter 7.4.

If the values for the monitoring times in the CPUs are different then the higher value always applies.

12.6.5 Step E: System transition to redundant mode

Initial situation

The redundant system operates with the modified CPU parameters in stand–alone mode.

Procedure

1. In SIMATIC Manager, select a CPU of the redundant system, then select the **PLC > Operating Mode** command.

2. From the **Operating Mode** dialog box, select the standby CPU, and then click **Restart (warm restart)**.

Result

The standby CPU will be coupled and updated. The redundant system is operating in redundant mode.

Reaction of the I/O

Type of I/O	One-sided I/O of standby CPU	One-sided I/O of master CPU	Switched I/O
I/O modules	Are reconfigured [1] and updated by the CPU.	Continue operation without interruption.	

1) The central modules will first be initialized with reset. Output modules briefly output 0 during this time (instead of the configured substitution or hold values).

Reaction to monitoring timeout

The update will be aborted if one of the monitored times exceeds the configured maximum. The redundant system remains in stand–alone mode with the previous master CPU, and under certain conditions retries to couple and update later. For further information, refer to chapter 7.4.

Where the values for the monitoring times in the CPUs differ, always the higher value always applies.

12.7 Modifying the CPU memory configuration

redundant state is only possible if both CPUs have the same memory configuration. Condition to be satisfied:

*   The size and type of load memory (RAM or FLASH) on both CPUs must match.

The memory configuration of the CPUs can be modified in run. Permissible modifications of S7-400H memory:

*   Expansion of load memory
*   Change of the type of load memory

12.7.1 Expanding load memory

The following methods of memory expansion are possible:

*   upgrade of the MMC with a card with more memory space
*   insertion of a RAM module if not already inserted

When these methods are deployed, the entire user program will be copied from the master CPU to the standby CPU during the coupling process (see chapter 7.3.1).

Restrictions

Memory should preferably be expanded using RAM modules, because this will ensure that the user program is copied to load memory of the standby CPU in the coupling process.

In principle, it is also feasible to use FLASH cards to expand load memory. However, you will then have to explicitly download the entire user program and the hardware configuration to the new FLASH card (see the procedure in chapter 12.7.2).

Initial situation

The redundant system is operating in redundant mode.

Procedure

Proceed in the following order:

Step	What Has To Be Done?	How Does the System React?
1	Switch the standby CPU to STOP using the PG.	The system is now operating in stand–alone mode.
2	Replace the memory card in the CPU with a card which has a higher capacity as required.	Standby CPU requests reset.
3	Reset the standby CPU using the PG.	–

Step	What Has To Be Done?	How Does the System React?
4	Start the standby CPU by means of the menu command **PLC > Mode > Switch to CPU with... expanded memory configuration**.	• The standby CPU is now coupled and updated, and assumes master mode. • Previous master CPU goes into STOP. • System operates in stand–alone mode.
5	Turn off power to the second CPU.	The unit is disabled.
6	Modify the memory configuration of the second CPU as you have done in step 2 to 3.	–
7	Start the second CPU using the PG.	• The second CPU will be coupled and updated. • The system is now operating in redundant mode again.

12.7.2 Changing the type of load memory

The following types of memory modules are available for load memory:

* RAM module for the test and commissioning phase
* FLASH card for the permanent storage of the completed user program

The size of the new memory card is irrelevant here.

When you deploy this method to modify your memory configuration, the system does not transfer any program elements from the master CPU to the standby CPU. Instead, it transfers only the contents of the unchanged blocks of the user program (see chapter 7.3.3).

It is the user's responsibility to download the entire user program to the new load memory.

Initial situation

The redundant system is operating in redundant mode.

The current status of the user program is available on the PG/ES as a STEP 7 project in modular form.

Caution

Here, you can not deploy a user program you uploaded from the PLC.

It is not permissible to recompile the user program from an STL source file, because this action would set a new time stamp at all blocks and thus prevent the block contents from being copied when you change over the master/standby station.

Procedure

Proceed in the following order:

Step	What Has To Be Done?	How does the system react?
1	Switch the standby CPU to STOP using the PG.	The system is now operating in stand–alone mode.
2	Replace the existing memory card in the standby CPU with a new one of the required type.	Standby CPU requests reset.
3	Reset the standby CPU using the PG.	–
4	Download the user program and the hardware configuration to the standby CPU.	–
5	Start the standby CPU by selecting the menu command **PLC > Mode > Switch to CPU with... modified configuration**.	• The standby CPU is now coupled and updated, and assumes master mode. • Previous master CPU goes into STOP. • The system now operates in stand–alone mode.
6	Modify the memory configuration of the second CPU as you did for the first CPU in step 2.	–
7	Download the user program and the hardware configuration to the partner CPU.	–
8	Start the second CPU using the PG.	• The second CPU will be coupled and updated. • The system is now operating again in redundant mode.

Notice

You can prepare the FLASH cards for a replacement by loading these with the CPU user program and hardware configuration on an external device and thus discard steps 4 and 7.

However, the memory cards in both CPUs must be loaded in the same sequence. Changing the order of blocks in the load memories will lead to a coupling failure.

Writing to a FLASH Card in the H System

You can always write access a FLASH card while the redundant system is in RUN, without having to stop the CPUs. The condition is, that the online data of the hardware configuration and the user program in both CPUs match the corresponding offline data in the engineering station.

Proceed as follows:

1. Set the standby CPU to STOP and insert the FLASH card into the CPU.

2. Perform a CPU memory reset with the help of STEP 7.

3. Download the hardware configuration using STEP 7.

4. Download the program data in STEP 7 by selecting the "Download User Program to Memory Card" command. Note: select the correct CPU from the selection dialog.

5. Switch to the CPU with the changed configuration using the "Operating Mode" dialog. This changes over the master/standby stations, and the CPU which contains the Flash card is now the master CPU. The standby CPU is now in STOP.

6. Next, insert the Flash card into the CPU which is in STOP. Perform a CPU memory reset using STEP 7.

7. Carry out step 4: Download the program data by selecting the "Download User Program to Memory Card" command in STEP 7. Note: select the correct CPU from the selection dialog.

8. Perform a warm restart of the standby CPU using the "Operating Mode" dialog. The system status now changes to "Redundant" mode.

The online and offline data consistency described earlier also applies when you remove FLASH Cards from a redundant system. In addition, the available RAM size may not be less than the actual size of the STEP 7 program (STEP 7 Program > Block Container > Properties "Blocks").

1. Set the standby CPU to STOP and remove the FLASH card. Adapt the RAM configuration as required.

2. Perform a CPU memory reset using STEP 7.

3. Download the block container using STEP 7.

4. Switch to the CPU with the changed configuration using the "Operating Mode" dialog.

5. Remove the FLASH card from the CPU which is now in STOP. Adapt the RAM configuration as required, and then perform a CPU memory reset.

6. Perform a warm restart of the standby CPU using the "Operating Mode" dialog. The system status now changes to "Redundant" mode.

12.8 Reconfiguration of a module

Refer to the information text in the "Hardware Catalog" window to determine which modules (signal modules and function modules) can be reconfigured during ongoing operation. The special reactions of individual modules are described in the respective technical documentation.

Notice

If you edit any protected parameters, the system will reject any attempt to changeover to the CPU containing those modified parameters. The error event W#16#5966 will be triggered and written to the diagnostics buffer, and you will then have to restore the previous valid values in the parameter configuration.

The selected new values must match the current and the planned user program.

Initial situation

The redundant system is operating in redundant mode.

Procedure

To edit the parameters of modules in a redundant system, perform the steps outlined below. Details of each step are listed in a subsection.

Step	What Has To Be Done?	Refer to Section
A	Editing parameters offline	12.8.1
B	Stopping the standby CPU	12.8.2
C	Downloading modified CPU parameters to the Standby CPU	12.8.3
D	Switch to the CPU which contains the modified configuration	12.8.4
E	System transition to redundant mode	12.8.5

Note

As of STEP 7 V5.3 SP2, after changing the hardware configuration, you can let the load operation run automatically. This means that you will not have to perform the action steps described in sections 12.8.2 to 12.8.5 anymore. The described system behaviour remains unchanged.

You can find more information in the online help HW Config **Download in module -> Download the station configuration in operating state RUN.**

12.8.1 Step A: Editing parameters offline

Initial situation

The redundant system is operating in redundant mode.

Procedure

1. Edit the module parameters offline in HW Config.

2. Compile the new hardware configuration, but do **not** download it into the PLC yet.

Result

The modified hardware configuration is available on the PG/ES. The PLC continues operation with the old configuration in redundant mode.

12.8.2 Step B: Stopping the standby CPU

Initial situation

The redundant system is operating in redundant mode.

Procedure

1. In SIMATIC Manager, select a CPU of the redundant system, and choose the **PLC > Operating Mode** menu command.

2. From the **Operating Mode** dialog box, select the standby CPU, then click **Stop**.

Result

The standby CPU goes into STOP, the master CPU remains in RUN. The redundant system now operates in stand–alone mode. One-sided I/O of the standby CPU are no longer addressed.

12.8.3 Step C: Downloading the new hardware configuration to the standby CPU

Initial situation

The redundant system is working in stand–alone mode.

Procedure

Download the compiled hardware configuration to the standby CPU while it is in STOP.

Notice

The user program and connection configuration may not be downloaded in stand–alone mode.

Result

The modified CPU parameters in the new hardware configuration of the standby CPU do not yet have an effect on the active process.

12.8.4 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 redundant system, and choose the **PLC > Operating Mode** menu command.

2. In the **Operating Mode** dialog box, click the **Toggle button.**

3. In the **Toggle** dialog box, select the option **with modified configuration** and click the **Toggle** button.

4. Confirm the query that follows with **OK**.

Result

The standby CPU links up, is updated and becomes the master. The former master CPU switches to STOP mode, the redundant system continues to work in stand–alone mode.

Behavior of the I/O

Type of I/O	One-sided I/O of previous master CPU	One-sided 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 reconfigured [1] and updated by the CPU.	Continue to work without interruption.

1) The central modules will first be initialized with reset. Output modules briefly output 0 during this time (instead of the substitution or hold values configured).

Reaction to monitoring timeout

The update will be aborted and no change of master takes place if one of the monitored times exceeds the configured maximum. The redundant system remains in stand–alone mode with the previous master CPU, and under certain conditions retries to couple and update later. For further information, refer to chapter 7.4.

If the values for the monitoring times in the CPUs are different then the higher value always applies.

Calling up the OB 83

After transfer of the parameter data records to the desired assemblies, OB 83 is started. The sequence is as follows:

1. After you have made the parameter changes to an module in STEP 7 and loaded them in RUN in the CPU, the OB 83 is started (trigger event W#16#3367). Relevant in the OB start information are the logical base address (OB83_MDL_ADDR) and the module type (OB83_MDL_TYPE). From now on, the input and/or initial data of the module might no longer be correct, and no SFCs that send data records to this module may be active.

2. After termination of OB 83, the parameters of the module are reset.

3. After termination of the parameter rest operation, the OB 83 is started again (trigger event W#16#3267 if the parameterization was successful, or W#16#3968 if it was unsuccessful). The input and/or initial data of the module is the same as after a 'connect' alarm, i.e. under certain circumstances it will not yet be correct. With immediate effect, you may start SFCs that send data records to the module.

12.8.5   Step E: Transition to redundant state

Initial situation

> The redundant system operates with the modified CPU parameters in stand–alone mode.

Procedure

> 1. In SIMATIC Manager, select a CPU of the redundant system, then select the **PLC > Operating Mode** command.
>
> 2. From the **Operating Mode** dialog box, select the standby CPU, and then click **Restart (warm restart)**.

Result

> The standby CPU will be coupled and updated. The redundant system is operating in redundant mode.

Reaction of the I/O

Type of I/O	One-sided I/O of standby CPU	One-sided I/O of master CPU	Switched I/O
I/O modules	Are reconfigured [1] and updated by the CPU.	Continue operation without interruption.	

[1] The central modules will first be initialized with reset. Output modules briefly output 0 during this time (instead of the configured substitution or hold values).

Reaction to monitoring timeout

> If one of the monitored times exceeds the configured maximum the update will be aborted. The redundant system remains in stand–alone mode with the previous master CPU, and under certain conditions retries to couple and update later. For further information, refer to chapter 7.4.
>
> Where the values for the monitoring times in the CPUs differ, always the higher value always applies.

Synchronization modules

<div style="text-align: right; font-size: 2em;">**13**</div>

Chapter Overview

13.1 Synchronization modules for S7-400H

Function of the synchronization modules

Synchronization modules areused for the communication between two redundant S7-400H CPUs. You require two synchronization modules per CPU. Connect these in pairs using fiber–optic cables.

The system supports hot–swapping of synchronization modules, and thus allows you to influence the repair reaction of the redundant system and to control the failure of the redundant connection without stopping the plant.

Distance between the S7-400H CPUs

Two types of synchronization modules are available:

Order Number	Maximum distance between the CPUs
6ES7960-1AA04-0XA0	10 m
6ES7960-1AB04-0XA0	10 km

Long synchronization cables may increase cycle times up to 10% per cable kilometer.

Notice

The redundant system requires four synchronization modules of the same type.

Mechanical Configuration

LED LINK OK for commissioning

Fiber–optic interface
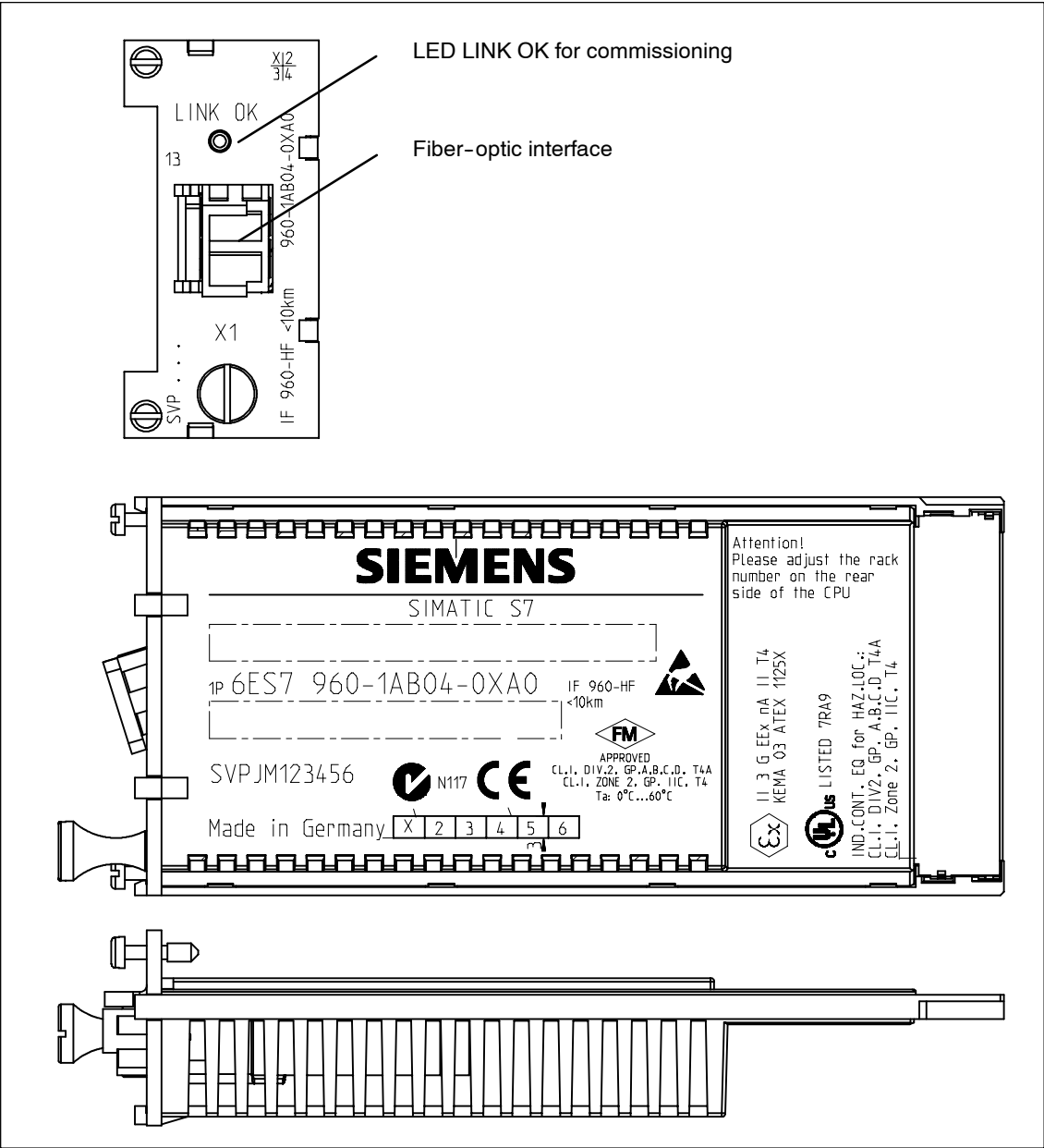
Figure 13-1   Synchronization Module

⚠ **Caution**

Risk of injury.

The synchronization module is equipped with a "CLASS 1 LASER PRODUCT" laser system to IEC 60825-1.

Avoid direct contact with the laser beam. Do not open the housing. Always observe the information provide in this manual, and keep this as reference.

> CLASS 1 LASER PRODUCT
> CLASS 1 LASER PRODUCT
> TO EN 60825

LED LINK OK

During commissioning of the redundant system, you can use the "LINK OK" LED on the synchronization module to check the quality of the connection between the CPUs.

LED LINK OK	Meaning
lit	The connection is OK
Flashes	The connection is not reliable, and the signal is disturbed Check the connectors and cables Check whether the fiber–optic cables are installed according to the guidelines of chapter 13.2
Dark	The connection is interrupted, or the light intensity is insufficient Check the connectors and cables Check whether the fiber–optic cables are installed according to the guidelines of chapter 13.2

OB84

When operating in redundant mode, the CPU operating system call OB84 if it detects a reduced performance in the redundant link between both CPUs.

Fiber–optic interfaces of unused modules

Fiber–optic interfaces of unused modules must be sealed during storage using the plugs which are supplied with the synchronization module..

Technical Specification

Technical Specifications	6ES7960-1AA04-0XA0	6ES7960-1AB04-0XA0
Maximum distance between the CPUs	10 m	10 km
Supply Voltage	5.1 V, supplied by the CPU	5.1 V, supplied by the CPU
Current consumption	210 mA	250 mA
Power loss	1,1 W	1.3 W
Wavelength of the optical transceiver	850 nm	1300 nm
Maximal permitted attenuation of the fiber–optic cable	7 dB	12 dB
Maximum permitted difference in cable lengths	9M	50M
Dimensions B x H x T (mm)	25 x 53 x 140	25 x 53 x 140
Weight	0.065 kg	0.065 kg

13.2   Installation of fiber‑optic cables

Introduction

Fiber–optic cables may only be installed by trained and qualified personnel. Always observe the current rules and legislation related to the safety of buildings. The installation must be carried out with meticulous care, because faulty installations represent the most common source of error. Causes are:

*   Kinking of the fiber–optic cable due to an insufficient bending radius.

*   Crushing of the cable as a result of excess forces caused by persons treading on the cable, or by pinching, or by the load of other heavy cables.

*   Overstretching due to high tensile forces.

*   Damage on sharp edges etc.

Permitted bending radius for prefabricated cables

You may not go below the bending radius when laying the cable:

*   Next to connector: 55 mm

*   During installation: 60 mm (repeated)

*   After installation: 40 mm (one-time)

To observe when installing the fiber‑optic cables for the S7-400H synchronization link

Always route the two fiber-optic cables separately. This increases availability, and protects the fiber-optic cables from potential double errors caused by simultaneous interruption.

Always make sure the fiber optic cables are connected to both CPUs before switching on the power supply or the system, for otherwise the CPUs may process the user program as the master CPU.

Local quality assurance

Check the items outlined below before you install the fiber-optic cables:

*   Does the delivered package contain the correct fiber-optic cables?

*   Any visible transport damage of the product?

*   Have you organized a suitable intermediate on–site storage for the fiber-optic cables?

*   Does the category of the cables match the connecting components?

Storage of the fiber-optic cables

if you do not install the fiber-optic cable immediately after you received the package, it is advisable to put it on shelf in a dry location where it is protected from mechanical and thermal influences. Observe the permitted storage temperatures specified in the data sheet of the fiber-optic cable. You should not remove the fiber-optic cables from the original package unless you are going to install them.

Open installation, wall ducts, cable ducts:

Observe the items outlined below when you install fiber-optic cables:

- The fiber-optic cables may be installed in open locations, provided you can safely exclude any damage in those areas (vertical risers, connecting ducts, telecommunication switchboard rooms, etc.).

- Fiber-optic cables should be mounted on mounting rails (cable trays, wire mesh ducts) using cable ties. Take care not to crush the cable when you fasten it (see pressure).

- Always deburr or round the edges of the duct before you install the fiber-optic cable, in order to prevent damage to the sheathing when you pull in and fasten the cable.

- The bending radii may not be smaller than the value specified in the manufacturer's data sheet.

- The branching radii of the cable ducts must correspond with the specified bending radius of the fiber-optic cable.

Pulling cables

Note the items below when you pull fiber-optic cables:

- Always observe the information on pull forces in the data sheet of the corresponding fiber-optic cable.

- Do not reel off any greater lengths when you pull in the cables.

- Install the fiber-optic cable directly from the cable drum, as far as possible.

- Do not spool the fiber-optic cable sideways off the drum flange (risk of twisting).

- You should use a cable pulling sleeve to pull in the fiber-optic cable.

- Always observe the specified bending radii.

- Do not use any fat– or oil–based lubricants.
  You may use the lubricants listed below to support the pulling of fiber-optic cables.

  – Yellow mass (Wire-Pulling, lubricant of Klein Tools; 51000)

  – soft soap

  – dishwashing liquid

  – talcum powder

  – washing agents

Pressure

Do not exert any pressure on the cable, for example, by the improper use of clamps (cable quick–mount) or cable ties. Your installation should also prevent anyone from stepping onto the cable.

Influence of heat

Fiber-optic cables are highly sensitive to direct heat, i.e. the cables may not be treated with hot–air guns or gas burners as used in heat–shrink tubing technology.

13.3   Selecting fiber‑optic cables

Make allowances for the following marginal conditions and situations when you select a suitable fiber-optic cable:

- Required cable lengths

- Indoor or outdoor installation

- Any particular protection against mechanical stress required?

- Any particular protection against rodents required?

- Installation of an outdoor cable directly under ground?

- Does the fiber-optic cable have to be water–proof?

- Which temperatures influence the installed fiber-optic cable?

Cable lengths up to 10 m

The synchronization module 6ES7 960-1AA04-0XA0 can be operated in pairs with fiber-optic cables up to a length of 10 m.

Select cables with the following specification for lengths up to 10 m:

- Multimode fiber 50/125 μ or 62.5/125 μ

- Patch cable for indoor applications

- 2 x duplex cable per H system, crossed over

- Connector type LC-LC

The following lengths of such cables are available as accessories for redundant systems

Table 13-1   Fiber-optic cable as accessory

Length	Order Number
1 m	6ES7960-1AA04-5AA0
2 m	6ES7960-1AA04-5BA0
10 m	6ES7960-1AA04-5KA0

Cable lengths up to 10 km

The synchronization module 6ES7 960-1AB04-0XA0 can be operated in pairs with fiber-optic cables up to a length of 10 km.

The following points are important:

* Make sure there is enough strain relief on the modules if you use fiber optic cables longer than 10 m.

* Keep to the specified ambient operating conditions of the fiber-optic cables used (bending radii, pressure, temperature...)

* Observe the technical specifications of the fiber optic cable (attenuation, bandwidth...)

Fiber-optic cables with lengths above 10 m usually have to be custom made. In the first step, select the following specification:

* Single–mode fiber (mono–mode fiber) 9/125 μ

   For short length required for testing and commissioning you may also use the lengths up to 10 m which are available as accessory. For continuous use, only the specified cables with monomode fibres are permitted.

The table below shows the further specifications, based on your application:

Table 13-2   Specification of fiber-optic cables for **indoor applications**

Cable routing	Components required	Specification
The cables are routed only within a building There is no cable junction between the indoor and outdoor area The necessary cable length is available in one piece. There is no need to connect several cable segments by means of distribution boxes. Complete installation using patch cables	Patch cables	• 2 x duplex cable per system • Connector type LC-LC • Crossed–over cables Further specifications you may need to observe for your plant: • UL certification • Halogen–free materials
	Patch cable	• Multicore cables, 4 conductors per system • Connector type LC-LC • Crossed–over cables Further specifications you may need to observe for your plant: • UL certification • Halogen–free materials

Table 13-2 Specification of fiber-optic cables for **indoor applications** , continued

Cable routing	Components required	Specification
The cables are routed only within a building There is no cable junction between the indoor and outdoor area The necessary cable length is available in one piece. There is no need to connect several cable segments by means of distribution boxes. Complete installation using patch cables	• , including patch cables for indoor applications as required	• 1 cable with 4 cores per redundant system Both interfaces in one cable • 1 or 2 cables with several shared cores Separate installation of the interfaces in order to increase availability (reduction of common cause factor) • Connector type ST or SC, for example, to match other components, see below Further specifications you may need to observe for your plant: • UL certification • Halogen–free materials Avoid spliced cables in the field. Use patch cables with pulling protection / tool whiplash or breakout design, including the measuring log.
	• Patch cable for indoor applications	• Connector type LC on ST or SC, for example, to match other components
Installation using distribution boxes, see Fig. 13-2	• One distribution/junction box per branch Installation cables and patch cables are interconnected by means of distribution box, using either ST or SC connectors, for example. Check the cross–over installation when you wire the CPUs.	• Connector type ST or SC, for example, to match other components

Table 13-3   Specification of fiber-optic cables for **outdoor applications**

Cable routing	Components required	Specification

The cabling requires a junction between the indoor and outdoor area
 	
see Figure 13-2
	• Installation cables for outdoor applications	Installation cables for outdoor applications
		• 1 cable with 4 cores per redundant system
		  Both interfaces in one cable
		• 1 or 2 cables with several shared cores
		  Separate installation of the interfaces in order to increase availability (reduction of common cause factor)
		• Connector type ST or SC, for example, to match other components, see below
		Further specifications you may need to observe for your plant:
		• UL certification
		• Halogen–free materials
		Observe further specifications as required for local conditions:
		• Protection against increased mechanical stress
		• Protection against rodents
		• Water–proofing
		• Suitable for direct underground installation
		• Suitable for the given temperature ranges
		Avoid spliced cables in the field. Use patch cables with pulling protection / tool whiplash design, including the measuring log.

	• including installation cables for indoor applications as required	• 1 cable with 4 cores per redundant system
		  Both interfaces in one cable
		• 1 or 2 cables with several shared cores
		  Separate installation of the interfaces in order to increase availability (reduction of common cause factor)
		• Connector type ST or SC, for example, to match other components, see below
		Further specifications you may need to observe for your plant:
		• UL certification
		• Halogen–free materials
		Avoid spliced cables in the field. Use patch cables with pulling protection / tool whiplash or breakout design, including the measuring log.

	• Patch cable for indoor applications	• Connector type LC on ST or SC, for example, to match other components

Table 13-3   Specification of fiber-optic cables for **outdoor applications** , continued

Cable routing	Components required	Specification
The cabling requires a junction between the indoor and outdoor area see Figure 13-2	• One distribution/junction box per branch Installation cables and patch cables are interconnected by means of distribution box, using either ST or SC connectors, for example. Check the cross–over installation when you wire the CPUs.	• Connector type ST or SC, for example, to match other components
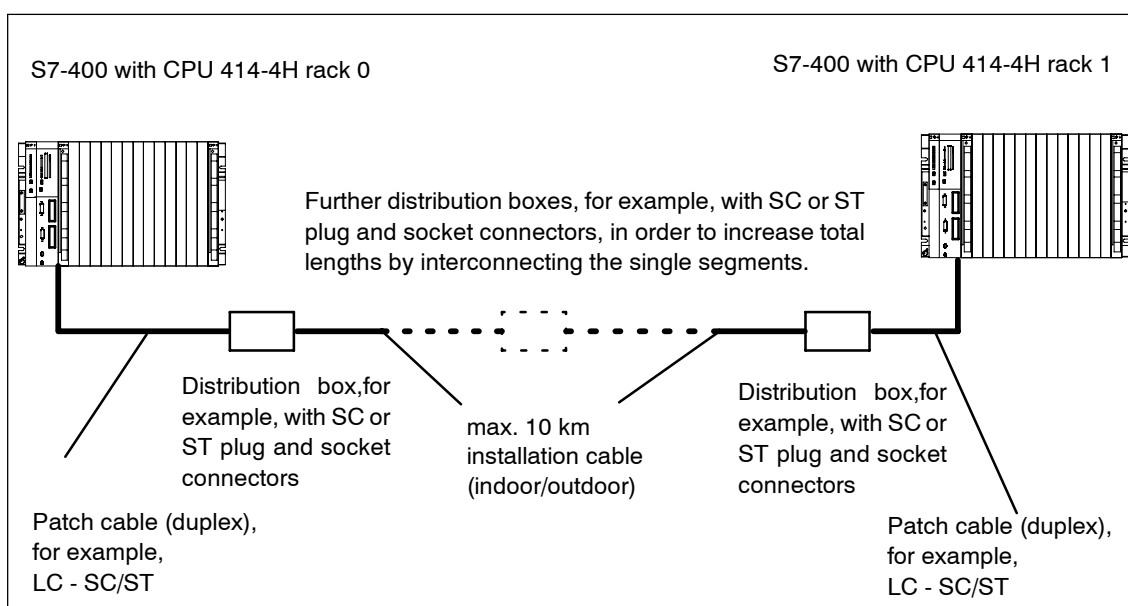
Figure 13-2   Fiber–optic cables, installation using distribution boxes

S7-400 cycle and reaction times

<div style="text-align: right; font-size: 3em; font-weight: bold;">14</div>

This chapter describes the decisive factors in the cycle and reaction times of your of S7-400 station.

You can read out the cycle time of the user program from the relevant CPU using the PG (see the *Configuring Hardware and Connections with STEP 7 V5.3* or higher).

The examples included show you how to calculate the cycle time.

An important aspect of a process is its reaction time. How to calculate this factor is described in detail in this chapter. When operating a CPU 41x-H as master on the PROFIBUS DP network, you also need to include the additional DP cycle times in your calculation (see chapter 14.5).

Chapter Overview

Further information

For further information on the execution times outlined below, refer to the S7-400 statement list. There you will the *STEP 7* statements supported by the relevant CPUs, and the integrated SFCs/SFBs or IEC functions with the relevant execution times you may call in *STEP 7* for each CPU.

14.1   Cycle time

This chapter describes the decisive factors in the cycle time, and how to calculate this time.

Definition of the cycle time

The cycle time defines the time the operating system requires to execute a program, i.e. to execute OB1, including all interrupt times required by program elements and for system activities.

This time is monitored.

Time slice model

The program, and thus the user program is executed cyclically in time slices. To demonstrate the processes, let us presume a global time slice length of 1 ms.

Process image

The CPU reads and writes the process signals to a process image before it starts cyclic program execution, in order to obtain a precise image of the process signals. The CPU does not access the signal modules directly when the I/O address areas respond during program execution, but rather addresses its memory area which contains the I/O process image.

Phases in the cyclic program execution

The table below shows the various phases in cyclic program execution.

Table 14-1   Cyclic program execution

Step	Sequence
1	The operating system initiates the scan cycle monitoring time.
2	The CPU writes the values of the process image to the outputs of the output modules.
3	The CPU reads the status of inputs of the input modules, and then updates the process image of inputs.
4	The CPU executes the user program in time slices, and executes the operations defined in the program.
5	At the end of the cycle, the operating system performs all pending tasks, such as loading or deleting blocks.
6	Finally, on expiration of any given minimum cycle time, the CPU returns to the start of the cycle and restarts scan cycle monitoring.

Elements of the cycle time
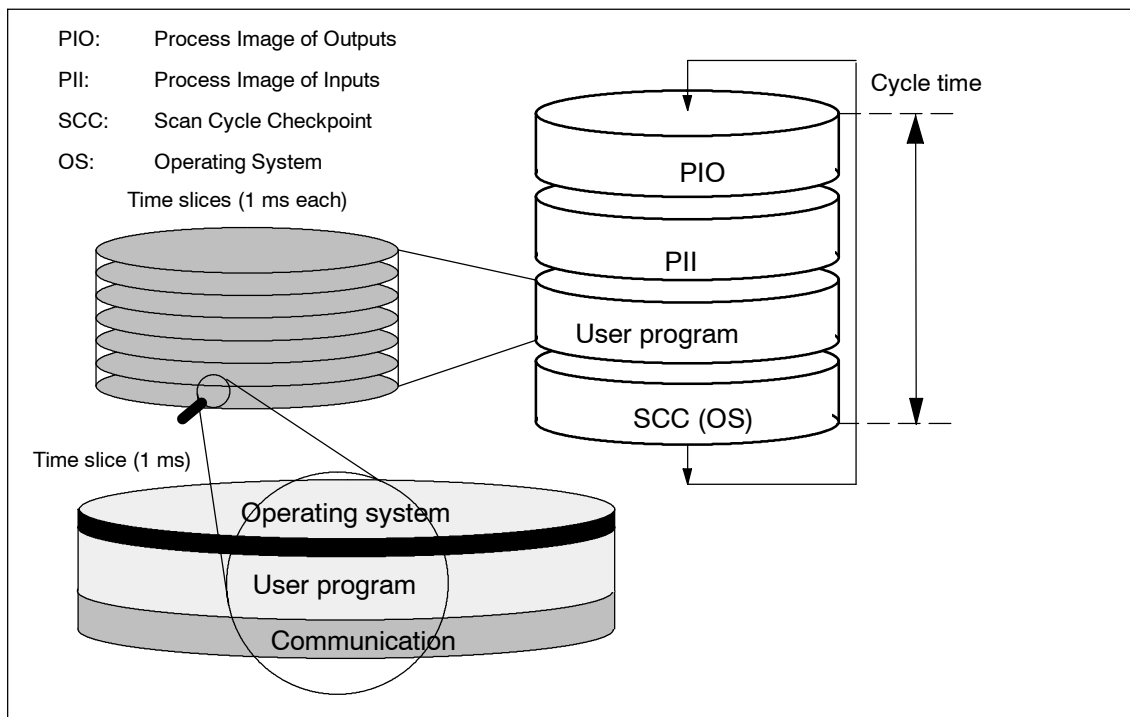
Figure 14-1   Elements and structure of the cycle time

14.2　Calculating the cycle time

Extension of the cycle time

The cycle time of a user program is extended by the factors outlined below:

- Time–based interrupt execution

- Process interrupt execution (see also chapter 14.8)

- Diagnostics and error processing (see also chapter 14.9)

- Communication via MPI and CPs connected to the communication bus (for example, Ethernet, PROFIBUS, DP) as a factor in communication load

- Special functions such as controlling and monitoring tags or the block status

- Download and deletion of blocks, compression of user program memory

Decisive factors

The table below shows the decisive factors in the cycle time.

Table 14-2　Decisive factors in the cycle time

Factors	Comment
Transfer time for the process image of outputs (PIO) and inputs (PII)	See table 14-5
User program execution time	This value is calculated based on the execution times of the various statements (see the *S7-400 statement list*). For special features of CPU 417-4H, refer to table 14-5.
Operating system execution time at the scan cycle checkpoint	See table 14-6
Extension of cycle times due to communication load	You configure the maximum permitted communication load on the cycle as a percentile value in *STEP 7* (*Programming without STEP 7*). See chapter 14.4.
Load on cycle times due to interrupts	Interrupt requests can always stop user program execution. See table 14-7

Process image update

The table below shows the times required by the CPU for a process image update (process image transfer time.) The specified times only represent "ideal values", and may be extended accordingly by any interrupts or communication of the CPU.

Calculation of the transfer time for process image updates:

C + portion in the PLC (taken from row A of the table below) + portion in the expansion device with local coupling (taken from row B) + portion in the expansion device with remote coupling (taken from row C) + portion of the integrated DP interface (taken from row D) + portion of consistent data via the integrated DP interface (taken from row E1) + portion of consistent data via external DP interface (taken from row E2)
= Transfer time for process image updates

The tables below contain the various portions of the transfer time for process image updates (process image transfer time)The specified times only represent "ideal values", and may be extended accordingly by any interrupts or communication of the CPU.

Table 14-3 Portion of the process image transfer time, CPU 414-4H

	Portions n = number of bytes in the process image m= number of accesses in the process image [*] k= number of consistent areas in the process image	CPU 414-4H stand–alone mode	CPU 414-4H redundant
K	base load	17 μs	17 μs
A [**]	In the CPU read byte/word/dword write byte/word/dword	$(m * 18 + n * 1,9)$ μs $(m * 11 + n * 1,9)$ μs	$(m * 26 + n * 1,9)$ μs $(m * 17 + n * 1,9)$ μs
B [**]	In the expansion device with local coupling read byte/word/dword write byte/word/dword	$(m *17 + n * 5)$ μs $(m * 11 + n * 5)$ μs	$(m * 26 + n * 5)$ μs $(m * 17 + n * 5)$ μs
C [**] [***]	In the expansion device with remote coupling read byte/word/dword write byte/word/dword	$(m * 14 + n * 12)$ μs $(m * 7 + n * 12)$ μs	$(m * 23 + n * 12)$ μs $(m * 13 + n * 12)$ μs
D	In the DP area for the integrated DP interface read byte/word/dword write byte/word/dword	$(m * 17 + n * 0,1)$ μs $(m * 11 + n * 0,1)$ μs	$(m * 30 + n * 0,1)$ μs $(m * 22 + n * 0,1)$ μs
I1	Consistent data in the process image for the integrated DP interface read data write data	$(k * 43 + n * 0,1)$ μs $(k * 37 + n * 0,1)$ μs	$(k * 175 + n * 0,6)$μs $(k * 155 + n * 0,1)$μs
I2	Consistent data in the process image for the external DP interface (CP 443-5 extended) read data write data	$(k * 41 + n * 3.4)$ μs $(k * 35 + n * 2)$ μs	$(k * 175 + n * 4)$μs $(k * 155 + n * 2)$μs

[*] The module data are updated with the minimum number of accesses.
(Example: 8 bytes result in 2 dword accesses, and 16 bytes in 4 dword accesses.)

[**] The specified value contains the cycle time of the I/O module which is inserted into the CPU or into an expansion device

[***] Measured with IM460-3 and IM461-3, at a coupling length of 100 m

Table 14-4   Portion of the process image transfer time, CPU 417-4H

	Portions n = number of bytes in the process image m= number of accesses in the process image *) k= number of consistent areas in the process image	CPU 417-4H stand–alone operation	CPU 417-4H redundant
K	base load	9 μs	10 μs
A **)	In the CPU read byte/word/dword write byte/word/dword	$(m * 10 + n * 1.8)$ μs $(m * 6 + n * 1.7)$ μs	$(m * 15 + n * 1.9)$ μs $(m * 9 + n * 1.8)$ μs
B **)	In the expansion device with local coupling read byte/word/dword write byte/word/dword	$(m * 10 + n * 5)$ μs $(m * 6 + n * 5)$ μs	$(m * 10 + n * 5)$ μs $(m * 6 + n * 5)$ μs
C **) ***)	In the expansion device with remote coupling read byte/word/dword write byte/word/dword	$m * 6 + n * 12$ μs $(m * 3 + n * 12)$ μs	$(m * 11 + n * 12)$ μs $(m * 6 + n * 12)$ μs
D	In the DP area for the integrated DP interface read byte/word/dword write byte/word/dword	$(m * 10 + n * 0.1)$ μs $(m * 6 + n * 0.1)$ μs	$(m * 16 + n * 0,1)$ μs $(m * 12 + n * 0,1$ μs
I1	Consistent data in the process image for the integrated DP interface read data write data	$(k * 25 + n * 0.1)$ μs $(k * 20 + n * 0.1)$ μs	$(k * 100 + n * 0.2)\mu$s $(k * 86 + n * 0.1)\mu$s
I2	Consistent data in the process image for the external DP interface (CP 443-5 extended) read data write data	$(k * 24 + n * 2.7)$ μs $(k * 22 + n * 1.8)$ μs	$(k * 100 + n * 3)\mu$s $(k * 91 + n * 1.9)\mu$s

*)   The module data are updated with the minimum number of accesses.
     (Example: 8 bytes result in 2 dword accesses, and 16 bytes in 4 dword accesses.)

**)   The specified value contains the cycle time of the I/O module which is inserted into the
     CPU or into an expansion device

***)   Measured with IM460-3 and IM461-3, at a coupling length of 100 m

Extension of cycle time at a 41x-4H CPU

The calculated cycle time of a 41x-4H CPU must be multiplied by a CPU–specific factor. The table below lists these factors:

Table 14-5   User program execution time of the 41x-4H CPU

Sequence	CPU414-4H stand–alone operation	CPU414-4H redundant	CPU 417-4H stand–alone operation	CPU 417-4H redundant
Factor	1.04	1.2	1.05	1, 2

Long synchronization cables may increase cycle times up to 10% per cable kilometer.

Operating system execution time at the scan cycle checkpoint

The table below shows the operating system execution time at the scan cycle checkpoint of the CPUs.

Table 14-6   Operating system execution time at the scan cycle checkpoint

Sequence	CPU 414H stand–alone operation	CPU 414H redundant	CPU 417-4H stand–alone operation	CPU 417-4H redundant
Cycle control at the SCC	301 - 1588 μs ∅ 317 μs	779 - 3262 μs ∅ 908 μs	163 - 1055 μs ∅ 214 μs	479 - 1765 μs ∅ 527 μs

Cycle time extension due to nested interrupts

Table 14-7   Cycle time extension due to nested interrupts

CPU	Process alarm	Diagnostics interrupt	Time–of–day interrupt	Delay interrupt	Watchdog interrupt	Program-ming / I/O access error	Asyn-chronous error
CPU 414-4H stand-alone operation	391 μs	406 μs	349 μs	269 μs	296 μs	147 μs / 167 μs	370 μs
CPU 414-4H redundant	768 μ	656 μs	730 μs	654 μs	702 μs	439 μs / 274 μs	1020 μs
CPU414-4H stand-alone operation	269 μs	279 μs	220 μs	185 μs	233 μs	89 μs / 102 μs	367 μs
CPU417-4H redundant	621 μs	563 μs	455 μs	415 μs	440 μs	246 μs / 149 μs	738 μs

Add the program execution time at interrupt level to this extension value.

The corresponding times will accumulate when the program contains nested interrupts.

14.3   Different cycle times

The length of cycle times (T_{cyc}) differs. The figure below shows the different cycle times T_{cyc1} and T_{cyc2}. T_{cyc2} is longer than T_{cyc1}, because the cyclically executed OB1 is interrupted by a TOD interrupt OB (here: OB 10).
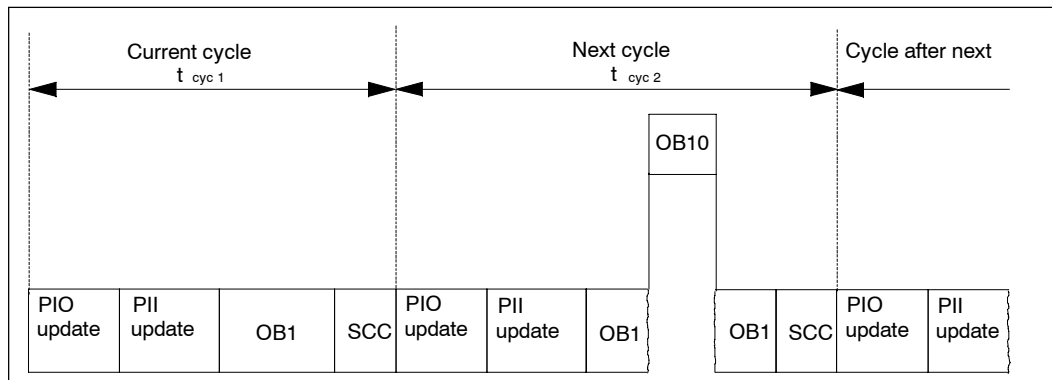
Figure 14-2   Different cycle times

A further factor in different cycle times is found in the variable block execution times (OB1, for example) caused by:

- conditional statements,

- conditional block calls,

- different program paths,

- loops etc.

Maximum cycle time

You can edit the default maximum cycle time (scan cycle monitoring time) in STEP 7. OB80 is called on expiration of this time. At this OB, you can define the CPU's reaction to the timeout error. Provided you do not retrigger the cycle time using SFC43, OB80 doubles the cycle time at its first call. The CPU will go into STOP at the second call of OB80.

CPU will go into STOP if OB80 does not exist in its memory.

Minimum cycle time

You can set the minimum CPU cycle time in STEP 7. This is useful if you

- want to set an interval of approximately the same length between the program execution cycles of OB1 (free cycle), or

- prevent unnecessary process image updates if the cycle time is too short
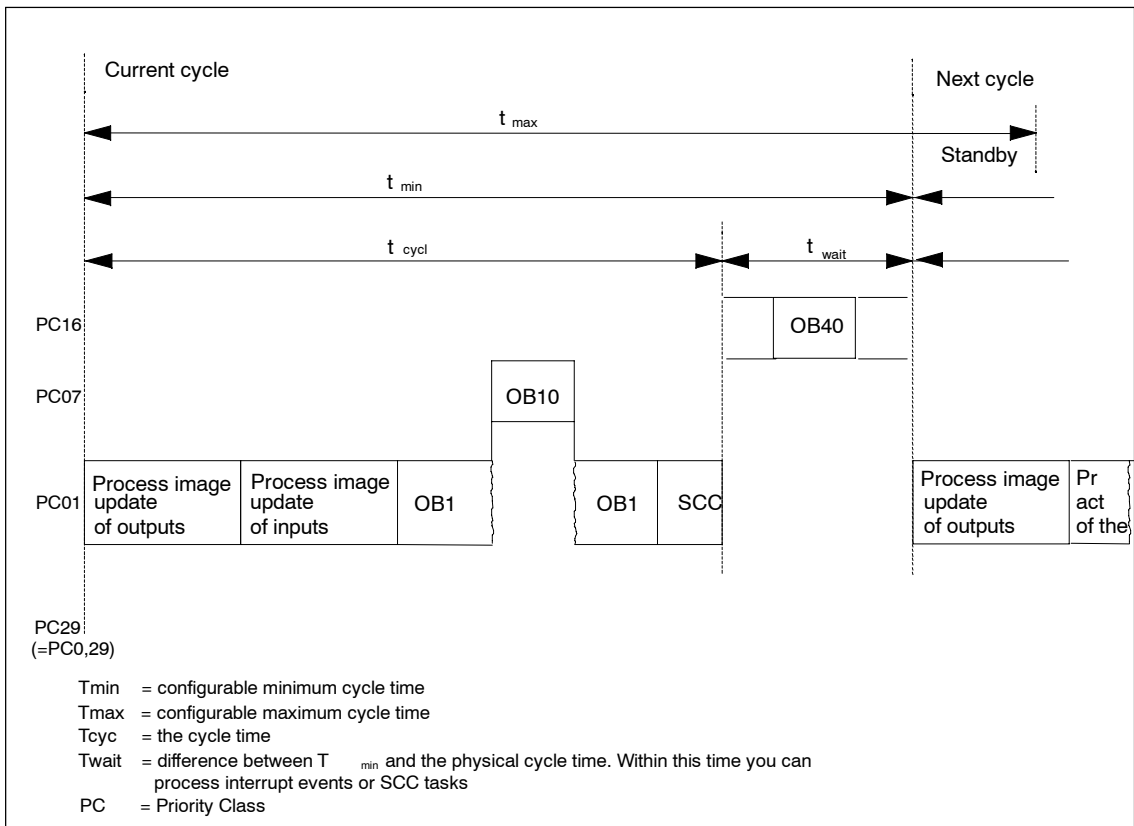
Figure 14-3   Minimum cycle time

The physical cycle time is derived from the sum of T_{cyc} and T_{wait}. It is thus always greater than or equal to T_{min}.

14.4   Communication load

The operating system provides the CPU continuously the configured time slices as a percentage of the overall CPU processing resources (time slice technique). If this processing capacity is not required for communication, it is made available to the other processes.

You can set a communication load between 5 % and 50 % in your hardware configuration. The default value is 20 %.

This percentage is to be interpreted as mean value, i.e. communication resources may take significantly more than 20 % of a time slice. Then again, the communication only takes a few or 0 % in the next time slice.

The formula below describes the influence of communication load on the cycle time:

$$\text{Physical cycle time} = \text{Cycle time} \times \frac{100}{100 - \text{"configured communication load in \%"}}$$

Round the result to the next higher integer !

Figure 14-4   Formula: Influence of communication load

Data consistency

The CPU interrupts the user program when it processes communication functions. This interruption can be triggered after any statement. Communication actions may lead to a change in user data, i.e. data consistency can not be ensured across several accesses.
How to ensure data consistency in operations comprising more than one instruction is described in the chapter Consistent data
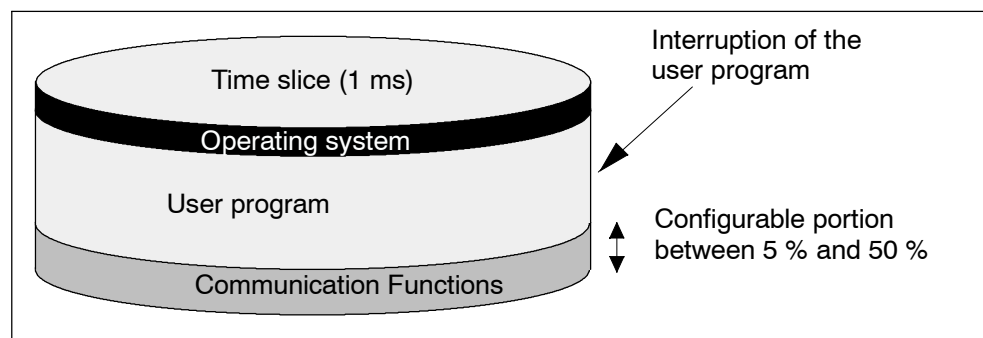
Figure 14-5   Distribution of a time slice

The operating system takes a certain portion of the remaining time slice for internal tasks. This portion isincluded in the factor defined in table 14-5 .

Example: 20 % communication load

In the hardware configuration, you have set a communication load of 20 %.

The calculated cycle time is 10 ms.

A setting of 20 % communication load thus allocates an average of 200 μs to communication and 800 μs to the user program in each time slice. The CPU thus requires 10 ms / 800 = 13 time slices to execute one cycle. The physical cycle time is therefore equivalent to 13 times 1–ms time slice = 13 ms, if the CPU fully utilizes the configured communication load.

Hence, 20 % communication does not extend the cycle by a linear factor of 2 ms, but rather by 3 ms.

Example: 50 % communication load

In the hardware configuration, you have set a communication load of 50 %.

The calculated cycle time is 10 ms.

The remaining resource for the cycle is thus reduced to 500 in each time slice. The CPU thus requires 10 ms / 500 = 20 time slices to execute one cycle. he physical cycle time is therefore equivalent to 20 ms, if the CPU fully utilizes the configured communication load.

A setting of 50 % communication load thus allocates an average of 500 μs to communication and 500 μs to the user program in each time slice. The CPU thus requires 10 ms / 500 = 20 time slices to execute one cycle. The physical cycle time is therefore equivalent to 20 times 1–ms time slice = 20 ms, if the CPU fully utilizes the configured communication load.

Hence, 20 % communication does not extend the cycle by a linear factor of 5 ms, but rather by 10 ms (= doubling of the calculated cycle time).

Dependency of the physical cycle time on communication load

The figure below describes the non–linear dependency of the physical cycle time on communication load. As an example we have selected a cycle time of 10 ms.
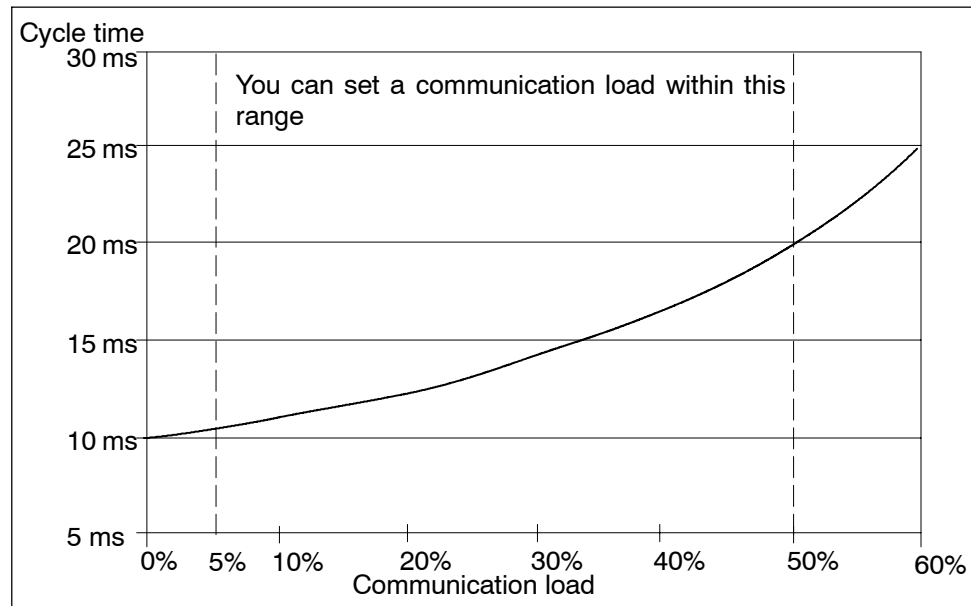
Figure 14-6   Dependency of the cycle time on communication load

Further effects on the physical cycle time

Statistics show that the extension of cycle times due to communication load leads to more asynchronous events than interrupts within an OB1 cycle, for example. This factor also extends the OB1 cycle. The extension is determined by the number of events per OB1 cycle and on the time required for processing these events.

Notes

- Check the effects on plant operation when you modify the value of the "Cycle load due to communication" parameter.

- Always make allowances for communication load when you set the maximum cycle time, for you would otherwise risk timeout errors.

Recommendations

- Apply the default value where possible.

- Increase this value only if the CPU is used primarily for communication, and if time is not a critical factor to the user program! You should only reduce this value in all other situations!

14.5   Reaction time

Definition of the reaction time

The reaction time represents the time expiring between the detection of an input signal and the modification of its logically linked output signal.

Fluctuation length

The physical reaction time lies between the shortest and longest reaction time. Always expect the longest reaction time when you configure your system.

The section below deals with the shortest and longest reaction times, in order to provide an overview of the fluctuation in the length of reaction times.

Factors

The reaction time is determined by the cycle time and the following factors:

- Delay at the I/Os

- Additional DP cycle times on the PROFIBUS DP network

- Processing in the user program

Delay of the I/Os

make allowances for the following module–specific delay times:

- the digital input delay times

- the input delay time +

     internal preparation times at digital inputs
  with interrupt function

- negligible delay times at digital outputs

- typical delay times of 10 ms to 20 ms at relay outputs.
     The delay of relay outputs is also determined
     by the temperature and voltage.

- The cycle time for analog input at analog inputs

- The output response time at analog outputs

For information on delay times, refer to the technical data of the signal modules.

DP cycle times on the PROFIBUS DP network

If you configured your PROFIBUS DP network in *STEP 7*, *STEP 7* calculates the typical DP cycle time to be expected. You can always view the DP cycle times of your configuration on the PG in the bus parameter section.

The figure below provides an overview of the DP cycle times. In this example, we assume an average value for each DP slave of four bytes of data.
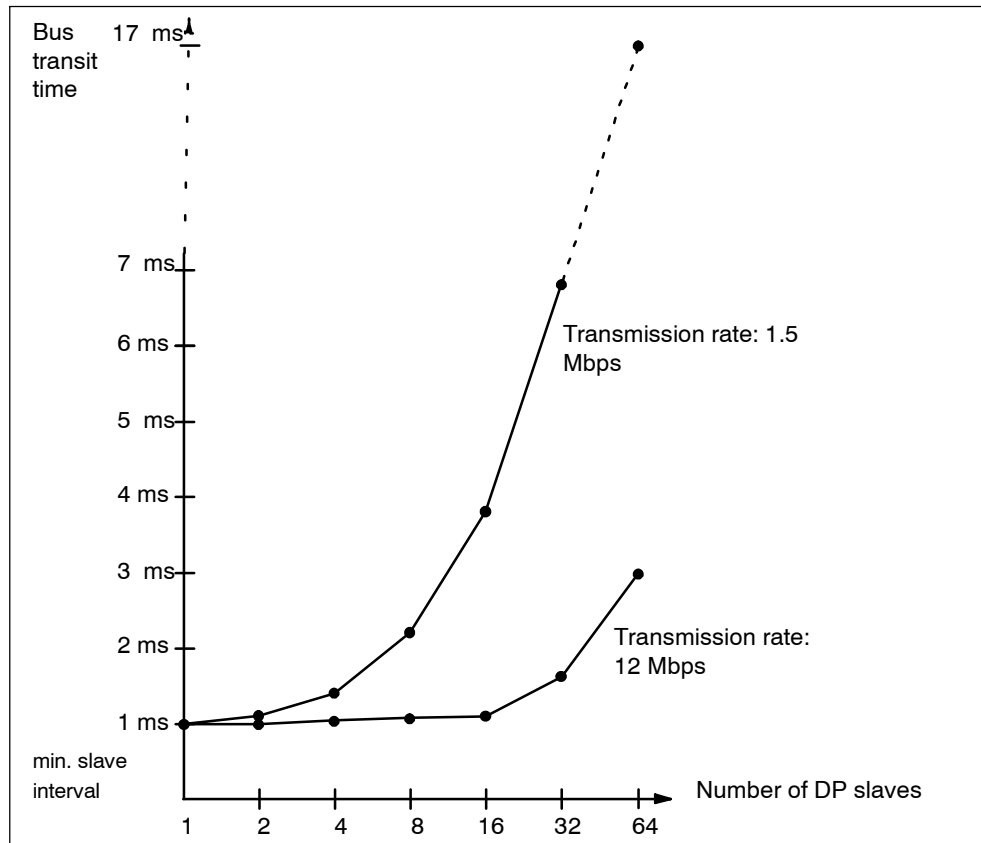
Figure 14-7   DP cycle times on the PROFIBUS DP network

If you are operating a PROFIBUS–DP network with more than one master, youmust take the DP cycle time into account for each master. In other words, perform a separate calculation for each master and add the results together..

Shortest reaction time

The following figure illustrates the conditions under which the shortest reaction time is achieved..
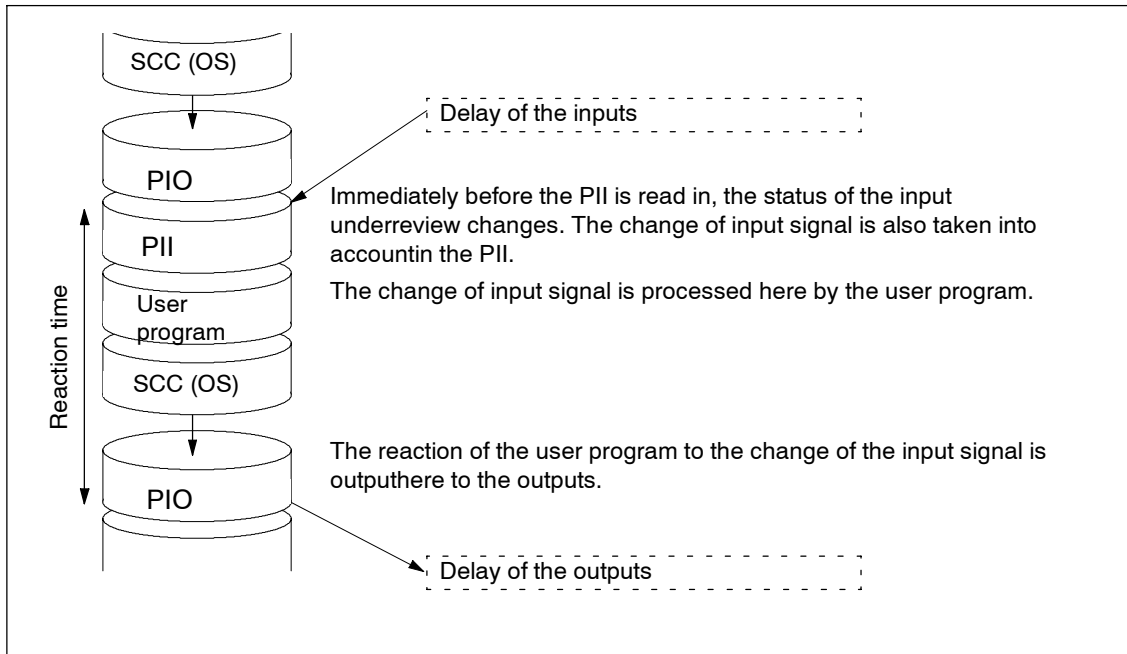
Figure 14-8   Shortest reaction time

Calculation

The (shortest) reaction time is made up as follows:

- 1 × process image transfer time of the inputs +

- 1 × process image transfer time of the outputs +

- 1 × program execution time +

- 1 × operating system processing time at SCC +

- Delay at inputs and outputs

This is equivalent to the sum of the cycle time and the delay at the inputs andoutputs.

Note

If the CPU and signal module are not in the central rack, you have to add double the runtime of the DP slave frame (includingprocessing in the DP master).

Longest reaction time

The following figure shows you how the longest reaction time results..
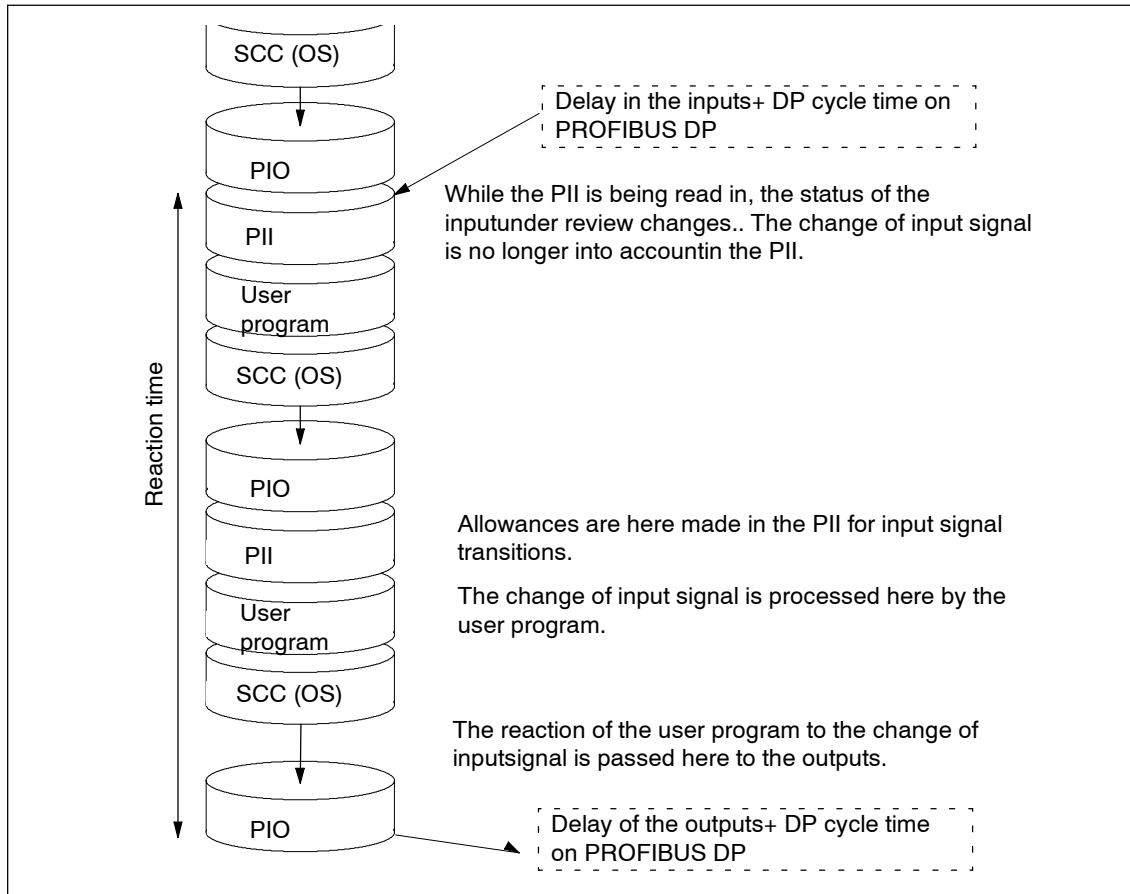
Figure 14-9   Longest reaction time

Calculation

The (longest) reaction time is derived from:

- 2 × process image transfer time of the inputs +
- 2 × process image transfer time of the outputs +
- 2 × program processing time +
- 2 × program execution time +
- 2 × runtime of the DP slave frame (including processing in the DP master) +
- Delay of inputs and outputs

This is equivalent to the sum of twice the cycle time and the delay in the inputs andoutputs plus twice the DP cycle time.

Direct I/O accesses

You can achieve faster reaction times by direct access to the I/O in the userprogram, for example with

- L PIB or
- T PQW.

you can work around the reaction times as shown earlier.

Reducing the reaction time

This the maximum reaction time to

- Delay at the inputs and outputs
- User program runtime (can be interrupted by high–priority interrupt handling)
- Runtime of direct accesses
- 2 x bus transit time of DP

The table below lists the execution times of direct accesses by the CPU to I/Omodules. The times shown are "ideal values"..

Direct accesses f the CPUs to I/O modules

Type of access	CPU 414-4H stand–alone operation	CPU 414-4H redundant	CPU 417-4H stand–alone operation	CPU 417-4H redundant
Read byte	33.3 µs	76.3 µs	20.4 µs	45.7 µs
Read word	36.2 µs	79.7 µs	22.6 µs	47.3 µs
Read dword	40.0 µs	84.4 µs	26.0 µs	50.6 µs
Write byte	32.4 µs	77.3 µs	19.9 µs	46.1 µs
Write word	35.4 µs	80.6 µs	22.0 µs	48.2 µs
Write dword	37.9 µs	83.6 µs	24.6 µs	50.6 µs

Direct accesses of the CPUs to I/O modules in the expansion device with local coupling

Type of access	CPU 41x-4H stand–alone operation	CPU 41x-4H redundant	CPU 41x-4 Hstand–alone operation	CPU 41x-4H redundant
Read byte	36.8 µs	78.9 µs	23.8 µs	46.9 µs
Read word	43.2 µs	82.9 µs	29.6 µs	52.7 µs
Read dword	54.0 µs	96.7 µs	40.0 µs	64.8 µs
Write byte	35.7 µs	79.1 µs	23.0 µs	47.2 µs
Write word	41.9 µs	83.6 µs	28.4 µs	52.5 µs
Write dword	51.0 µs	96.0 µs	37.6 µs	63.5 µs

Direct accesses of the CPUs to I/O modules in the expansion device with remote coupling

Type of access	CPU 41x-4H stand–alone operation	CPU 41x-4H redundant	CPU 41x-4H stand–alone operation	CPU 41x-4H redundant
Read byte	40.5 μs	80.2 μs	27.4 μs	50.5 μs
Read word	53.3 μs	93.3 μs	39.8 μs	62.8 μs
Read dword	77.3 μs	120.0 μs	63.4 μs	88.0 μs
Write byte	39.2 μs	81.0 μs	26.7 μs	50.7 μs
Write word	52.1 μs	93.9 μs	38.7 μs	62.7 μs
Write dword	74.3 μs	119.5 μs	60.9 μs	86.8 μs

Note

The specified times are merely CPU processing times and apply, unless otherwisestated, to signal modules in the central rack.

You can similarly achieve fast reaction times by using hardware interrupts, see chapter 14.8.

14.6   Calculating cycle and reaction times

Cycle time

1. Using the Instruction List, determine the runtime of the user program.

2. Calculate and add the transfer time for the process image. You will find guidevalues for this in the tables 14-3 and 14-4.

3. Add to it the processing time at the scan cycle checkpoint. You will find guidevalues for this in table 14-6.

4. Multiply the calculated value by the factor in table 14-5.

The result is the **cycle time**.

Extension of the cycle time due to communication and interrupts

5. Multiply the result by the following factor:

$$\frac{100}{100 - \text{"configured communication load in \%"}}$$

6. Using the instruction list, calculate the runtime of the program elements processing the interrupts. Add to it the relevant value in table 14-7.
    Multiply this value by the factor from step 4.
    Add this value to the theoretical cycle time as often as the interrupt is triggered or is expected to be triggered during the cycle time.

The result you obtain is approximately the **actual cycle time**. Make a note of theresult.

Table 14-8   Example of calculating the reaction time

Shortest reaction time	Longest reaction time
7. Next, calculate the delays in theinputs and outputs and, if applicable,the DP cycle times on the PROFIBUS DP network.	7. Multiply the actual cycle time by thefactor 2.
	8. Next, calculate the delays in theinputs and outputs andthe DP cycle times on the PROFIBUS DP network.
8. The result you obtain is the **shortestreaction time.**	9. The result you obtain is the **longestreaction time.**

14.7 Examples of calculating the cycle time and reactiontime

Example I

You have installed an S7–400 with the following modules in the central rack:

- a 414-4H CPU in redundant mode

- 2 digital input modules SM 421; DI 32×DC 24 V (each with 4 bytes in the PI)

- 2 digital output modules SM 422; DO 32×DC 24 V /0.5(each with 4 bytes in the PI)

User program

According to the Instruction List, your user program has a runtime of 15 ms.

Calculating the cycle time

The cycle time for the example is derived from the following factors:

- The CPU-specific factor is 1.2. Thus, the approximate user program execution time is:
  18.0 ms

- Process image transfer time

  Process image: 26 μs + 16 Byte×1,4 μs = ca. **0.05 ms**

- OS execution time at the scan cycle checkpoint:
  approx. **0.96 ms**

The cycle time is equivalent to the sum of the listed times:

Cycle time = 18.0 ms + 0.05 ms + 0.96 ms **= 19.01 ms.**

Calculating the actual cycle time

- Allowance for communication load (default value: 20%):
  19.01 ms * 100 / (100-20) **= 23.76 ms.**

- There is no interrupt handling..

The rounded actual cycle time is thus **24 ms.**

Calculating the longest reaction time

- Longest reaction time24.0ms * 2 **= 48.0 ms.**

- The delay of the I/O is negligible.

- All the components are installed in the central rack. DP cycle times can thus be ignored.

- There is no interrupt handling..

The rounded longest reaction time is thus = **48 ms.**

Example II

You have installed an S7–400 with the following modules in the central rack:

- a 414-4H CPU in redundant mode

- 4 digital input modules SM 421; DI 32×DC 24 V (each with 4 bytes in the PI)

- 3 digital output modules SM 422; DO 16×DC 24 V /2 (each with 2 bytes in the PI)

- 2 analog input modules SM 431; AI 8×13 bit (not in the PI)

- 2 analog output modules SM 432; AO 8×13 bit (not in the PI)

CPU parameters

The CPU has been assigned parameters as follows:

- Cycle load due to communication: 40 %

User program

According to the Instruction List, your user program has a runtime of 10.0 ms.

Calculating the cycle time

The theoretical cycle time for the example is derived from the following factors:

- The CPU-specific factor is 1.2. The approximate user program execution time is thus:
    12.0 ms

- Process image transfer time

    Process image: 17 μs + 22 Byte×1.4 μs = ca. **0.05 ms**

- OS runtime at the SCC:
    approx. **0.96 ms**

The cycle time is equivalent to the sum of the listed times:

   Cycle time = 12.0 ms + 0.05 ms + 0.96 ms = **13.01 ms.**

Calculating the actual cycle time

- Allowance for communication load:
    13.01 ms * 100 / (100-40) = **21.7 ms.**

- A time–of–day interrupt having a runtime of 0.5 ms is triggered every 100 ms.
    The interrupt can not be triggered more than once during a cycle:
    0.5 ms + 0.81 ms (se table 14-7) = **1.31 ms.**
    Allowance for communication load :
    1.31 ms * 100 / (100-40) = **2.18 ms.**

- 21.7 ms + 2.18 ms = **23.88 ms.**

Making allowances for the time slices, the actual cycle time is thus **24 ms.**

Calculation of the longest reaction time

- Longest reaction time
    24 ms * 2 = **48 ms**

- Delay of inputs and outputs

    - The maximum input delay of the digital input module SM 421; DI 32×DC 24 V is **4.8 ms** per channel

    - The delay of the digital output module SM 422; DO 16×DC 24 V/2A is negligible.

    - Analog input module SM 431; AI 8×13 bits was configured for 50 Hz interference frequency suppression. This results in a conversion timeof 25 ms per channel. The cycle time of the analog output module is equivalent to **200 ms**, because all eight channels are active.

    - Analog output module SM 432; AO 8×13 bits is configured for operation in the measuring range 0 V to 10V. This results in a conversion timeof **0.3** ms per channel. The cycle time is equivalent to 2.4 ms, because all eight channels are active. Add the transient time of a resistive load of 0. ms. The result is an analog output response time of **2.5 ms**.

- All the components are installed in the central rack. DP cycle times can thus be ignored.

- **Use case 1:** The system sets a digital output channel after a digital signal is read in. The result is a reaction time of

    = 48 ms+ 4.8 ms **= 52.8 ms**.

- **Use case 2:** The system reads and outputs an analog value. The result is a reaction time of

    = 48 ms+ 200 ms + 2.5 ms **= 250.5 ms**.

14.8   Interrupt reaction time

Definition of the interrupt reaction time

The interrupt reaction time is equivalent to the interval expiring between an incoming interrupt signal and the call of the first instruction in the interrupt OB.

General rule: Interrupts having a higher priority takeprecedence, i.e. the interrupt reaction time is increased by theprogram execution time of the higher priority interrupt OBs, and by interrupt OBs ofthe same priority which have not yet been processed (queue).

Note
Read/write requests with a maximum data volume (approx. 460 bytes) may delay the interrupt action times.
The system currently supports only the transfer of diagnostics interrupts **or** process alarms between the CPU and DP master at any given time on the DP segment.

Calculation

	Min. interrupt reaction time of the CPU		Max. interrupt reaction time of the CPU
+	minimum interrupt reaction time of the signal modules	+	maximum interrupt reaction time of the signal modules
+	DP cycle time on PROFIBUS DP		+ 2 * DP cycle time on PROFIBUS DP
=	Shortest interrupt reaction time	=	Longest interrupt reaction time

Figure 14-10   Calculation of the interrupt reaction time

Process alarm and diagnostics interrupt reaction times of the CPUs

Table 14-9   Process alarm and diagnostic interrupt reaction times; maximum interrupt reaction time without communication

CPU	Process alarm reaction times		Diagnostic interrupt reaction times	
	min.	**max.**	**min.**	**max.**
414-4Hstand–alonemode	330 µs	571 µs	351 µs	591 µs
414-4Hredundant	617 µs	951 µs	470 µs	940 µs
417-4Hstand–alone mode	207 µs	351 µs	218 µs	356 µs
417-4Hredundant	378 µs	645 µs	291 µs	776 µs

Extension of the maximum interrupt reaction time due to communication load

The maximuminterrupt reaction time increases when communication functions areactive. The increase is calculated with the following formula:

CPU 41x-4H $\quad t_v = 100\ \mu s + 1000\ \mu s \times n\%$, a significant extension may be expected

whereby n = cycle load due to communication

Signal modules

The process alarm reaction time of signal modules is derived from:

- Digital input modules

    Process alarm reaction time = internal interrupt processing time + input delay

    For information on times, refer to the data sheet of the relevant digital input module.

- Analog input modules

    Process alarm reaction time = internal interrupt processing time + conversion time

    The internal interrupt processing time of the analog input modules is negligible. For information on conversion times, refer to the data sheet of the relevant analog input module.

The diagnostic interrupt reaction time of the signal modules is equivalent to the time expiring between the detection of a diagnostics event by the signal module, and thetriggering of the diagnostic interrupt by this module. This slight time factor may be ignored.

Process alarm processing

Process alarm processing is initiated with the call of process alarm OB4x. Interrupts of higher priority interrupt process alarm processing, and directaccess to the I/O is made when the instruction is executed. After the process alarm has been processed, the system either resumes cyclic program execution, or calls and processes interrupt OBs of the same or lower priority.

14.9 Example of the calculation of the interrupt reaction time

Elements of the interrupt reaction time

As a reminder: the process alarm reaction time is derived from:

- the process alarm reaction time of the CPU, and

- the process alarm reaction time of the signal module.

- 2 × DP cycle time on PROFIBUS DP

Example: You have installed a 417-4H CPU and four digital modules in the central rack. One digital input module is the SM 421; DI 16⊕UC 24/60 V, with process alarms and diagnostic interrupts. In the CPU and SM parameters, you have only enabled the process alarm. You waved time–driven processing, diagnostics and error handling, and configured an input delay of 0.5 ms for the digital input module. Any actions at the scan cyclecheckpoint are not required. You have set a communication load of 20 % for the cycle.

Calculation

The process alarm reaction time for the example is derived from the following factors:

- Process alarm reaction time of the 417-4H CPU: approx. 0.5ms (mean value in redundant mode)

- Extension due to communication load according to the formula shown in the footer of table 14-9 :

$$100\ \mu s + 1000\ \mu s \times 20\% \ = \ 300\ \mu s = \ 0.3\ ms$$

- Process alarm reaction time of SM 421; DI 16 x UC 24/60 V:

  - Internal interrupt processing time: 0.5 ms

  - Input delay: 0.5 ms

- The DP cycle timeon the PROFIBUS–DP is irrelevant, because the signal modules are installed in the central rack.

The process alarm reaction time is equivalent to the sum of the listed times:

Process alarm reaction time =
 0.5 ms + 0.3 ms + 0.5 ms + 0.5 ms = approx.**1.8 ms**.

This calculated process alarm reaction time is equivalent to the time expiring between the detection of a signal at the digital input and the call of the first statement in OB4x.

14.10   Reproducibility of delay and watchdog interrupts

Definition of "Reproducibility"

Delay interrupt:

The time–based difference between the call of the first statement of the interrupt OB and the programmed time of interrupt.

Watchdog interrupt:

The fluctuation of the time–based interval between two successive calls, measured between each initial statement of the interrupt OB.

Reproducibility

Table 14-10 lists the reproducibility of delay and watchdog interrupts of theCPUs.

Table 14-10 Reproducibility of delay and watchdog interrupts of theCPUs

Module	Reproducibility	
	Delay interrupt:	**Watchdog interrupt**
414-4H CPU stand–alone mode	**–592 μs / +642 μs**	**–495 μs / +490μs**
CPU 414-4H redundant	**–558 μs / +780 μs**	**–842 μs / +699 μs**
CPU 417-4H stand–alone operation	**–548 μs / +418 μs**	**–385 μs / +385 μs**
CPU 417-4H redundant	**–362 μs / +626 μs**	**–410 μs / +380 μs**

These times apply only if the system is actually able to execute the interrupt at this point in time, without any delays, for example, due to higher–priority interrupts or pending interrupts of the same priority.

Technical Specifications

15

Chapter Overview

15.1 Technical Specifications of the CPU 414-4H; (6ES7 414-4HJ04-0AB0)

CPU and Version	
• Hardware version	6ES7 414-4HJ04-0AB0
• Firmware version	4.0 V
Associated programming package	As of STEP7 5.2 SP1 HF3 with HW-Update
Memory	
Working memory	
• Integrated	700 Kbytes for code
	700 Kbytes for data
Load memory	
• Integrated	256 Kbytes RAM
• Expandable FEPROM	With memory card (FLASH) 1 Mbyte up to 64 Mbytes
• Expandable RAM	With memory card (RAM) 256 Kbytes up to 16 Mbytes
Backup with battery	Yes, all data
Processing Times	
Processing times for	
• Bit operations	0.06 μs
• Word instructions	0.06 μs
• Integer math instructions	0.06 μs
• Floating-point math instructions	0.18 μs
Timers/Counters and Their Retentivity	
S7 counters	2048
• Retentivity can be set	From Z 0 to Z 2047
• Preset	From Z 0 to Z 7
• Counting range	1 to 999
IEC counter	Yes
• Type	SFB
S7 timers	2048
• Retentivity can be set	From T 0 to T 2047
• Preset	No retentive timers
• Time range	10 ms to 9990 s
IEC timers	Yes
• Type	SFB

Data Areas and Their Retentivity	
Total retentive data area (incl. memory markers, timers, counters)	Total working and load memory (with backup battery)
Memory markers	8 Kbytes
• Retentivity can be set	From MB 0 to MB 8191
• Preset retentivity	From MB 0 to MB 15
Clock memories	8 (1 memory byte)
Data blocks	Max. 4095 (DB 0 reserved)
• Size	Max. 64 Kbytes
Local data (can be set)	Max. 16 Kbytes
• Preset	8 Kbytes
Blocks	
OBs	See instruction list
• Size	Max. 64 Kbytes
Nesting depth	
• Per priority class	24
• Additionally in an error OB	1
FBs	Max. 2048
• Size	Max. 64 Kbytes
FCs	Max. 2048
• Size	Max. 64 Kbytes
Address Areas (Inputs/Outputs)	
Total I/O address area	8 Kbytes/8 Kbytes
• Of which distributed	incl. diagnostics addresses, addresses for I/O interfaces, etc
MPI/DP interface	2 Kbytes/2 Kbytes
DP interface	6 Kbytes/6 Kbytes
Process Image	8 Kbytes/8 Kbytes (can be set)
• Preset	256 bytes/256 bytes
• Number of partial process images	Max. 15
• Consistent data	Max. 244 bytes
Digital channels	Max. 65536/Max. 65536
• those central	Max. 65536/Max. 65536
Analog channels	Max. 4096/Max. 4096
• those central	Max. 4096/Max. 4096

Configuration	
Central racks/expansion units	Max. 1/21
Multicomputing	No
Number of plug-in IMs (overall)	Max. 6
• IM 460	Max. 6
• IM 463-2	maximum 4, only in single operation
Number of DP masters	
• Integrated	2
• Via CP 443–5 ext.	Max. 10
Operable function modules and communication processors	
• FM, see Appendix E	Limited by the number of slots and the number of connections
• CP 441	Limited by the number of connections (maximum of 30)
• Profibus and Ethernet CPs including CP 443-5 Extended	Max. 14

Time	
Clock	Yes
• Buffered	Yes
• Resolution	1 ms
• Accuracy at	
– Power off	Max. deviation per day: 1.7 s
– Power on	Max. deviation per day: 8.6 s
Runtime meter	8
• Number	0 to 7
• Value Range	0 to 32767 hours
• Granularity	1 hour
• Retentive	Yes
Time synchronization	Yes
• In PLC, on MPI and DP	as master or slave

S7 Message Functions	
Number of stations that can log on for message functions (e.g. WIN CC or SIMATIC OP)	Max. 8
Block-related messages	Yes
• Simultaneously active ALARM_S/SQ blocks and ALARM_D/DQ blocks	Max. 100
ALARM_8 blocks	Yes
• Number of communication jobs for ALARM_8 blocks **and** blocks for S7 communication (can be set)	Max. 1200
• Preset	900
Statuses	Yes
Number of archives that can log on simultaneously (SFB 37 AR_SEND)	16

Test and Startup Functions	
Monitor/modify tag	Yes
• Variables	Inputs/outputs, memory markers, DB, distributed inputs/outputs, timers, counters
• Number of variables	Max. 70
Force	Yes
• Variables	Inputs/outputs, memory markers, distributed inputs/outputs
• Number of variables	Max. 256
Status block	Yes
Single sequence	Yes
Diagnostics buffer	Yes
• Number of entries	Max. 3200 (can be set)
• Preset	120
Number of breakpoints	4

Communication Functions	
Programming device/OP communication	Yes
Number of connectable OPs	31 without message processing
	8 with message processing
Number of connection resources for S7 connections via all interfaces and CPs	32, with one each of those reserved for PG and OP
S7 communication	Yes
• User data per job	Max. 64 Kbytes
– Of which consistent	1 variable (462 bytes)
S7 basic communication	no
Global data communication	no
S5-compatible communication	via FC AG_SEND and AG_RECV, max. via 10 CP 443-1 or 443-5
• User data per job	Max. 8 Kbytes
– Of which consistent	240 bytes
Standard communication (FMS)	yes (via CP and loadable FB)

Interfaces
You must **not** configure the CPU as DP slave.

1st Interface	
Type of interface	Integrated
Physical	RS 485/Profibus
Isolated	Yes
Power supply to interface (15 VDC to 30 VDC)	Max. 150 mA
Number of connection resources	MPI: 32 DP: 16

Functionality	
• MPI	Yes
• PROFIBUS DP	DP Master

1st Interface MPI mode	
• Utilities	
– Programming device/OP communication	Yes
– Routing	Yes
– S7 communication	Yes
– Global data communication	No
– S7 basic communication	No
• Transmission rates	Up to 12 Mbps

1st Interface DP Master mode	
• Utilities	
– Programming device/OP communication	Yes
– Routing	Yes
– S7 communication	Yes
– Global data communication	No
– S7 basic communication	No
– Equidistance	No
– SYNC/FREEZE	No
– Enable/disable DP slaves	No
• Transmission rates	Up to 12 Mbps
• Number of DP slaves	Max. 32
• Address area	Max. 2 Kbytes inputs/ 2 Kbytes outputs
• User data per DP slave	Max. 244 bytes inputs, max.244 bytes outputs, max. 244 slots each with max. 128 bytes

Note:
- The accumulated number of input bytes at the slots may not exceed 244.
- The accumulated number of output bytes at the slots may not exceed 244.
- The maximum address area of the interface (max. 2 KB inputs/2 KB outputs) accumulated by 32 slaves may not be exceeded.

2nd Interface	
Type of interface	Integrated
Physical	RS 485/Profibus
Isolated	Yes
Power supply to interface (15 VDC to 30 VDC)	Max. 150 mA
Number of connection resources	16

Functionality	
• PROFIBUS DP	DP Master

2nd Interface DP Master mode	
• Utilities	
– Programming device/OP communication	Yes
– Routing	Yes
– S7 communication	Yes
– Global data communication	No
– S7 basic communication	No
– Equidistance	No
– SYNC/FREEZE	No
– Enable/disable DP slaves	No
• Transmission rates	Up to 12 Mbps
• Number of DP slaves	Max. 96
• Address area	Max. 6 Kbytes inputs/6 Kbytes outputs
• User data per DP slave	Max. 244 bytes inputs, max.244 bytes outputs, max. 244 slots each with max. 128 bytes

Note:

- The accumulated number of input bytes at the slots may not exceed 244.
- The accumulated number of output bytes at the slots may not exceed 244.
- The maximum address area of the interface (max. 2 KB inputs/2 KB outputs) accumulated by 32 slaves may not be exceeded.

3rd Interface	
Type of interface	Plug-in synchronisation module (fiber-optic cable)
Insertable interface submodule	Synchronization module IF 960 (only during redundancy mode; during single mode the interface is free/covered)

4th Interface	
Type of interface	Plug-in synchronisation module (fiber-optic cable)
Insertable interface submodule	Synchronization module IF 960 (only during redundancy mode; during single mode the interface is free/covered)

Programming	
Programming language	LAD, FBD, STL, SCL
Instruction set	See instruction list
Bracket levels	8
System functions (SFC)	See instruction list

Number of SFCs active at the same time for every strand	
• RD_REC	8
• WR_REC	8
• WR_PARM	8
• PARM_MOD	1
• WR_DPARM	2
• DPNRM_DG	8
• RDSYSST	1 ... 8
• DP_TOPOL	1

The accumulated number of active SFCs on all external strands can be four times as many as on one single strand.

System function blocks (SFC)	See instruction list

Number of SFBs active at the same time	
• RD_REC	8
• WR_REC	8

The accumulated number of active SFBs on all external strands can be four times as many as on one single strand.

User program protection	Password protection
Access to consistent data in the process image	Yes

CiR synchronization time (in single mode)	
Base load	100 ms
Time per I/O byte	100 μs

Dimensions	
Mounting dimensions B×H×T (mm)	50×290×219
Slots required	2
Weight	Approx. 1.1 kg

Voltages, Currents	
Current consumption from S7-400 bus (5 VDC)	Typ. 1.5 A Max. 1.7 A
Current consumption from the S7-400 bus (24 VDC) The CPU does not consume any current at 24 V, and it only makes this voltage available at the MPI/DP interface.	Total current consumption of the components connected to the MPI/DP interfaces, with a maximum of 150 mA per interface
Backup current	Typ. 550 μA Max. 1530 μA
maximum backup time	76 days
Incoming supply of external backup voltage to the CPU	5 VDC to 15 VDC
Power loss	Typ. 7.0 W

15.2 Technical Specifications of the CPU 417-4H; (6ES7 417-4HL04-0AB0)

CPU and Version	
MLFB	6ES7 417-4HL04-0AB0
• Firmware version	1.0 V
Associated programming package	As of STEP7 5.2 SP1 HF3 with HW-Update
Memory	
Working memory	
• Integrated	10 Mbytes for code 10 Mbytes for data
Load memory	
• Integrated	256 Kbytes RAM
• Expandable FEPROM	With memory card (FLASH) 1 Mbyte up to 64 Mbytes
• Expandable RAM	With memory card (RAM) 256 Kbytes up to 16 Mbytes
Backup with battery	Yes, all data
Processing Times	
Processing times for	
• Bit operations	0.03 μs
• Word instructions	0.03 μs
• Integer math instructions	0.03 μs
• Floating-point math instructions	0.09 μs
Timers/Counters and Their Retentivity	
S7 counters	2048
• Retentivity can be set	From Z 0 to Z 2047
• Preset	From Z 0 to Z 7
• Counting range	1 to 999
IEC counter	Yes
• Type	SFB
S7 timers	2048
• Retentivity can be set	From T 0 to T 2047
• Preset	No retentive timers
• Time range	10 ms to 9990 s
IEC timers	Yes
• Type	SFB

Data Areas and Their Retentivity	
Total retentive data area (incl. memory markers, timers, counters)	Total working and load memory (with backup battery)
Memory markers	16 Kbytes
• Retentivity can be set	From MB 0 to MB 16383
• Preset retentivity	From MB 0 to MB 15
Clock memories	8 (1 memory byte)
Data blocks	Max. 8191 (DB 0 reserved)
• Size	Max. 64 Kbytes
Local data (can be set)	Max. 64 Kbytes
• Preset	32 Kbytes
Blocks	
OBs	See instruction list
• Size	Max. 64 Kbytes
Nesting depth	
• Per priority class	24
• Additionally in an error OB	2
FBs	Max. 6144
• Size	Max. 64 Kbytes
FCs	Max. 6144
• Size	Max. 64 Kbytes
Address Areas (Inputs/Outputs)	
Total I/O address area	16 Kbytes/16 Kbytes
• Of which distributed	incl. diagnostics addresses, addresses for I/O interfaces, etc
MPI/DP interface	2 Kbytes/2 Kbytes
DP interface	8 Kbytes/8 Kbytes
Process Image	16 Kbytes/16 Kbytes (can be set)
• Preset	1024 bytes/1024 bytes
• Number of partial process images	Max. 15
• Consistent data	Max. 244 bytes
Digital channels	Max. 131072/Max. 131072
• those central	Max. 131072/Max. 131072
Analog channels	Max. 8192/Max. 8192
• those central	Max. 8192/Max. 8192

Configuration	
Central racks/expansion units	Max. 1/21
Multicomputing	No
Number of plug-in IMs (overall)	Max. 6
• IM 460	Max. 6
• IM 463-2	maximum 4 in single operation
Number of DP masters	
• Integrated	2
• Via CP 443-5 ext.	Max. 10
Number of plug-in S5 modules via adapter casing (in the central rack)	None
Operable function modules and communication processors	
• FM, see Appendix E	Limited by the number of slots and the number of connections
• CP 441	Limited by the number of connections, maximum of 30
• Profibus and Ethernet CPs including CP 443-5 Extended	Max. 14

Time	
Clock	Yes
• Buffered	Yes
• Resolution	1 ms
• Accuracy at	
– Power off	Max. deviation per day: 1.7 s
– Power on	Max. deviation per day: 8.6 s
Runtime meter	8
• Number	0 to 7
• Value Range	0 to 32767 hours
• Granularity	1 hour
• Retentive	Yes
Time synchronization	Yes
• In PLC, on MPI and DP	as master or slave

S7 Message Functions	
Number of stations that can log on for message functions (e.g. WIN CC or SIMATIC OP)	Max. 16
Block-related messages	Yes
• Simultaneously active ALARM_S/SQ blocks and ALARM_D/DQ blocks	Max. 200
ALARM_8 blocks	Yes
• Number of communication jobs for ALARM_8 blocks **and** blocks for S7 communication (can be set)	Max. 10000
• Preset	1200
Statuses	Yes
Number of archives that can log on simultaneously (SFB 37 AR_SEND)	64

Test and Startup Functions	
Monitor/modify tag	Yes
• Variables	Inputs/outputs, memory markers, DB, distributed inputs/outputs, timers, counters
• Number of variables	Max. 70
Force	Yes
• Variables	Inputs/outputs, memory markers, distributed inputs/outputs
• Number of variables	Max. 512
Status block	Yes
Single sequence	Yes
Diagnostics buffer	Yes
• Number of entries	Max. 3200 (can be set)
• Preset	120
Number of breakpoints	4

Communication Functions	
Programming device/OP communication	Yes
Number of connectable OPs	63 without message processing
	16 with message processing
Number of connection resources for S7 connections via all interfaces and CPs	64, with one each of those reserved for PG and OP
S7 communication	Yes
• User data per job	Max. 64 Kbytes
– Of which consistent	1 variable (462 bytes)
S7 basic communication	no
Global data communication	no
S5-compatible communication	via FC AG_SEND and AG_RECV, max. via 10 CP 443-1 or 443-5
• User data per job	Max. 8 Kbytes
– Of which consistent	240 bytes
Standard communication (FMS)	Yes (via CP and downloadable FC)

Interfaces
You must **not** configure the CPU as DP slave.

1st Interface	
Type of interface	Integrated
Physical	RS 485/Profibus
Isolated	Yes
Power supply to interface (15 VDC to 30 VDC)	Max. 150 mA
Number of connection resources	MPI: 44 DP: 32, a diagnostic repeater in the strand reduces the number of connection resources by 1

Functionality	
• MPI	Yes
• PROFIBUS DP	DP Master

1st Interface MPI mode	
• Utilities	
– Programming device/OP communication	Yes
– Routing	Yes
– S7 communication	Yes
– Global data communication	no
– S7 basic communication	no
• Transmission rates	Up to 12 Mbps

1st Interface DP Master Mode	
• Utilities	
– Programming device/OP communication	Yes
– Routing	Yes
– S7 communication	Yes
– Global data communication	No
– S7 basic communication	No
– Equidistance	No
– SYNC/FREEZE	No
– Enable/disable DP slaves	No
• Transmission rates	Up to 12 Mbps
• Number of DP slaves	Max. 32
• Address area	Max. 2 Kbytes inputs/ 2 Kbytes outputs
• User data per DP slave	Max. 244 bytes inputs, max.244 bytes outputs, max. 244 slots each with max. 128 bytes/slot

Note:

- The accumulated number of input bytes at the slots may not exceed 244.
- The accumulated number of output bytes at the slots may not exceed 244.
- The maximum address area of the interface (max. 2 KB inputs/2 KB outputs) accumulated by 32 slaves may not be exceeded.

2nd Interface	
Type of interface	Integrated
Physical	RS 485/Profibus
Isolated	Yes
Power supply to interface (15 VDC to 30 VDC)	Max. 150 mA
Number of connection resources	32, a diagnostic repeater in the strand reduces the number of connection resources by 1

Functionality	
• PROFIBUS DP	DP Master

2nd Interface DP Master Mode	
• Utilities	
– Programming device/OP communication	Yes
– Routing	Yes
– S7 communication	Yes
– Global data communication	No
– S7 basic communication	No
– Equidistance	No
– SYNC/FREEZE	No
– Enable/disable DP slaves	No
• Transmission rates	Up to 12 Mbps
• Number of DP slaves	Max. 125
• Address area	Max. 8 Kbytes inputs/ 8 Kbytes outputs
• User data per DP slave	Max. 244 bytes inputs, max.244 bytes outputs, max. 244 slots each with max. 128 bytes

Note:
- The accumulated number of input bytes at the slots may not exceed 244.
- The accumulated number of output bytes at the slots may not exceed 244.
- The maximum address area of the interface (max. 2 KB inputs/2 KB outputs) accumulated by 32 slaves may not be exceeded.

3rd Interface	
Type of interface	Plug-in synchronisation module (fiber-optic cable)
Insertable interface submodule	Synchronization module IF 960 (only during redundancy mode; during single mode the interface is free/covered)

4th Interface	
Type of interface	Plug-in synchronisation module (fiber-optic cable)
Insertable interface submodule	Synchronization module IF 960 (only during redundancy mode; during single mode the interface is free/covered)

Programming	
Programming language	LAD, FBD, STL, SCL
Instruction set	See instruction list
Bracket levels	8
System functions (SFC)	See instruction list

Number of SFCs active at the same time for every strand	
• RD_REC	8
• WR_REC	8
• WR_PARM	8
• PARM_MOD	1
• WR_DPARM	2
• DPNRM_DG	8
• RDSYSST	1 ... 8
• DP_TOPOL	1

The accumulated number of active SFCs on all external strands can be four times as many as on one single strand.

System function blocks (SFB)	See instruction list

Number of SFCs active at the same time	
• RD_REC	8
• WR_REC	8

The accumulated number of active SFBs on all external strands can be four times as many as on one single strand.

User program protection	Password protection
Access to consistent data in the process image	Yes

CiR synchronization time (in single mode)	
Base load	100 ms
Time per I/O byte	50 μs

Dimensions	
Mounting dimensions B×H×T (mm)	50×290×219
Slots required	2
Weight	Approx. 1.1 kg

Voltages, Currents	
Current consumption from S7-400 bus (5 VDC)	Typ. 1.5 A Max. 1.7 A
Current consumption from the S7-400 bus (24 VDC) The CPU does not consume any current at 24 V, and it only makes this voltage available at the MPI/DP interface.	Total current consumption of the components connected to the MPI/DP interfaces, with a maximum of 150 mA per interface
Backup current	Typically 600 μA Maximum 1810 μA
maximum backup time	71 days
Incoming supply of external backup voltage to the CPU	5 VDC to 15 VDC
Power loss	Typ. 7.0 W

15.3 Run Times of the FCs and FBs for Redundant I/O

Table 15-1   Run times of the blocks for redundant I/O

Block	Run time in single/single mode	Run time in redundant mode
FC 450 RED_INIT Specifications are based on the startup	2 ms + 300 μs / configured module pairs The specification for a module pair is a mean value. The run time may be < 300 μs for a few modules. For the great majority of redundant modules the value may even be > 300 μs.	–
FC 451 RED_DEPA	160 μs	360 μs
FB 450 RED_IN Invoked from the corresponding sequence level.	750 μs + 60 μs / module pair of the current TPA The specification for a module pair is a mean value. The run time may be additionally increased if discrepancies occur resulting in passivation and logging to the diagnostics buffer. The run time may also be increased by a depassivation carried out in the individual sequence levels of FB RED_IN. Depending on the number of modules in the sequence level, the depassivation may increase the run time of the FB RED_IN by 0.4 to 8 ms. An 8 ms increase can be expected in redundant operation of modules totalling more than 370 pairs of modules in a sequence level.	1000 μs + 70 μs/ module pair of the current TPA The specification for a module pair is a mean value. The run time may be additionally increased if discrepancies occur resulting in passivation and logging to the diagnostics buffer. The run time may also be increased by a depassivation carried out in the individual sequence levels of FB RED_IN. Depending on the number of modules in the sequence level, the depassivation may increase the run time of the FB RED_IN by 0.4 to 8 ms. An 8 ms increase can be expected in redundant operation of modules totalling more than 370 pairs of modules in a sequence level.
FB 451 RED_OUT Invoked from the corresponding sequence level.	650 μs + 2 μs / module pair of the current TPA The specification for a module pair is a mean value. The run time may be < 2 μs for a few modules. For the great majority of redundant modules the value may even be > 2 μs.	860 μs + 2 μs / module pair of the current TPA The specification for a module pair is a mean value. The run time may be < 2 μs for a few modules. For the great majority of redundant modules the value may even be > 2 μs.

Table 15-1   Run times of the blocks for redundant I/O, continued

Block	Run time in single/single mode	Run time in redundant mode
FB 452 RED_DIAG	Invoked in OB 72: 160 μs Invoked in OB82, 83, 85: 250 μs + 5 μs / configured module pairs Under extreme conditions the run time of FB RED_DIAG is increased up to 1.5 ms. This is the case when the working DB is 60 Kb or larger and if there are interrupt trigger addresses that do not belong to the redundant I/O.	Invoked in OB 72: 360 μs Invoked in OB82, 83, 85: 430 μs (basic load) + 6 μs / configured module pairs Under extreme conditions the run time of FB RED_DIAG is increased up to 1.5 ms. This is the case when the working DB is 60 Kb or larger and if there are interrupt trigger addresses that do not belong to the redundant I/O.
FB 453 RED_STATUS	160 μs + 4 μs/ configured module pairs * total number of module pairs The run time depends on the (random) position of the module being searched for in the working DB. When a module address is not redundant, the entire working DB is searched. This results in the longest run time of FB RED_STATUS. The number of module pairs is based either on all inputs (DI/AI) or all outputs (DO/AO).	350 μs + 5 μs/ configured module pairs * total number of module pairs) The run time depends on the (random) position of the module being searched for in the working DB. When a module address is not redundant, the entire working DB is searched. This results in the longest run time of FB RED_STATUS. The number of module pairs is based either on all inputs (DI/AI) or all outputs (DO/AO).

Notice

These are typical and not absolute values. The actual value may deviate from these specifications in some cases. This overview is intended as a guide and should help you estimate how the use of the RED_IO library may change the cycle time.

Parameters of redundant automation systems

<div style="float: right; font-size: 3em; font-weight: bold;">A</div>

This appendix provides a brief introduction to the parameters of redundant automation systems, and shows the practical effects of redundant configurations, based on a selection of configurations.

An overview of the MTBF of various SIMATIC products is available in the SIMATIC FAQs at
`http://www.siemens.com/automation/service&support`
under the ID 1160399.

In chapter	you will find	on page
A.1	Basic concepts	A-2
A.2	Comparison of the MTBF of selected configurations	A-7

A.1   Basic concepts

The quantitative assessment of redundant automation systems is usually based on their reliability and availability parameters. These are described in detail below.

Reliability

Reliability refers to the capability of technical equipment to fulfil its function during its operating period. This is usually not the case if any of its components fails.

The commonly set criterion for reliability is therefore the **MTBF** (**M**ean **T**ime **B**etween **F**ailures). This can be analyzed statistically based on the parameters of live systems, 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 extensive quality assurance measures in design and production.

Reliability of automation systems

The use of redundant modules considerably prolongs the MTBF of a system. The combination of integrated high-quality self-tests and error detection mechanisms of the S7-400H CPUs allows the detection and localization of virtually all errors.

The MTBF of an S7-400H is determined by the mean down time **MDT** (**M**ean **D**own **T**ime) of a system unit. This time is derived in essence from the error detection time plus the time required to repair or replace defective modules.

In addition to other measures, a CPU provides a self–test function with an adjustable test cycle time. The default test cycle time is 90 minutes. This time has an influence on the error detection time. The repair time usually required for a modular system such as the S7-400H is four hours.

Mean Down Time (MDT)

The MDT of a system is determined by the times outlined below:

*   time required to detect an error
*   time required to find the cause of an error
*   time required for troubleshooting and to restart the system

The system MDT is calculated based on the MDT of the various system components, including the components which form the system.

Association between MDT and MTBF: MDT << MTBF

The MDT value is of highest significance for the quality of system maintenance. The most important factors are:

*   Qualified personnel
*   Efficient logistics

- High–performance tools for diagnostics and error recognition
- A solid repair strategy

The figure below shows the dependency of the MDT on the times and factors mentioned earlier.
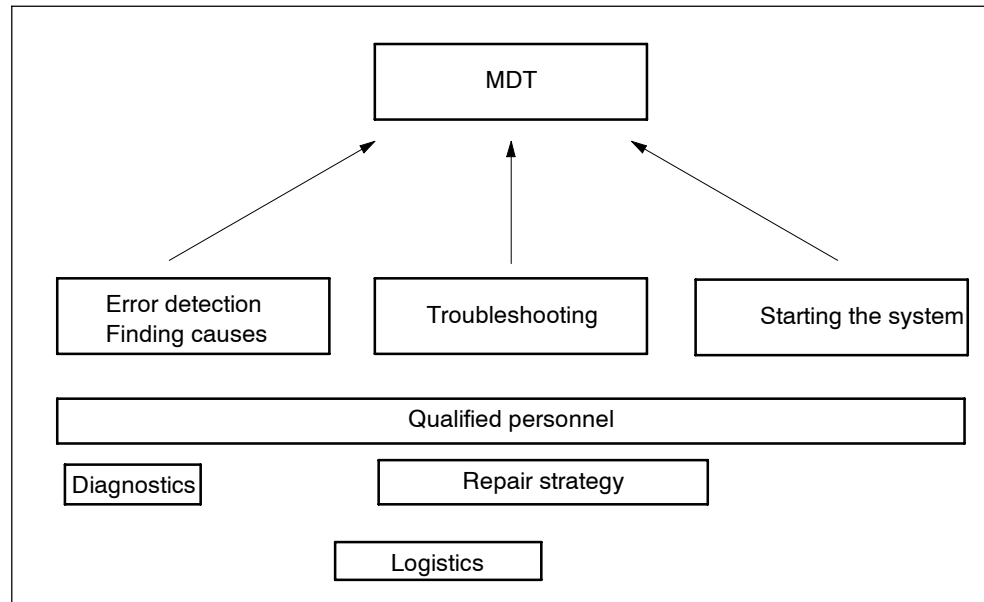
Figure A-1   MDT

The figure below shows the parameters included in the calculation of the MTBF of a system.
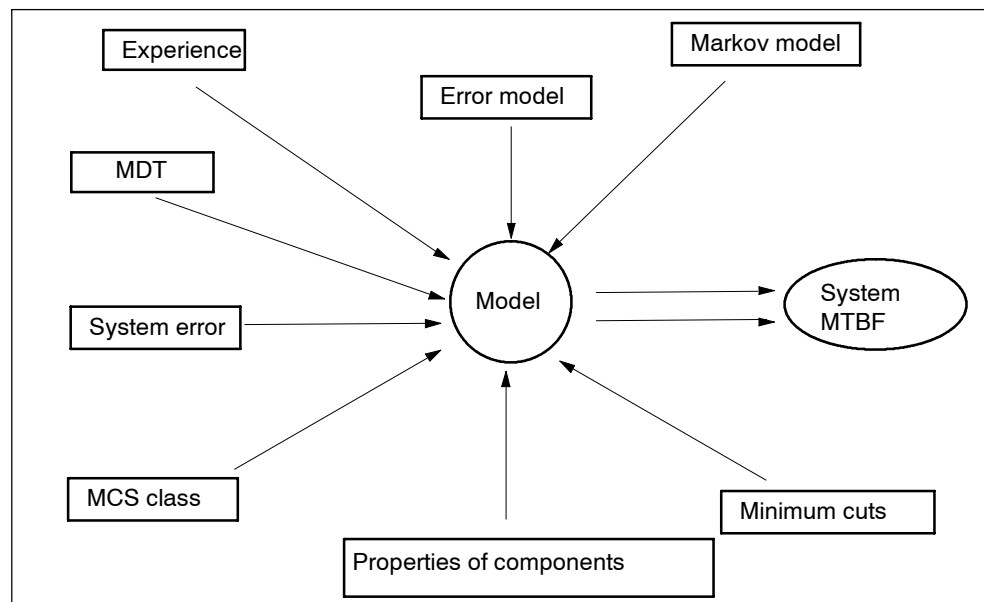
Figure A-2   MTBF

Requirements

This analysis assumes the following conditions:

- The error rate of all components and all calculations are based on an average temperature of 40 °C.

- The system installation and configuration is free of errors.

- All replacement parts are available locally, in order to prevent extended repair times due to missing spare parts. This keeps the component MDT down to minimum.

- The MDT of the various components is four hours. The system's MDT is calculated based on the MDT of the various components plus the system structure.

- The MTBF of the components is compliant with the SN 29500 standard. This standard corresponds with MIL-HDBK 217-F.

- The calculations are made using the diagnostics data of each component.

- Assumed is a CCF factor between 0.2 % and 2 %, based on the system configuration.

Common Cause Failure (CCF)

The Common Cause Failure (CCF) is an error which is caused by one or several events which also lead to an error state at two or more separate channels or components in a system. A CCF leads to a system failure.

The CCF may be cause by one of the following factors:

- – temperature
- – humidity
- – corrosion
- – vibration and shock
- – EMC load
- – electrostatic discharge
- – RF interference
- – unexpected sequence of events
- – faulty operator input

The CCF factor defines the ratio between the probability of the occurrence of a CCF and the probability of the occurrence of any other error.

Typical CCF factors range from 2% to 0. % in a system with identical components, and between 1% and 0.1% in a system containing different components.

Within the range stipulated in IEC 61508, a CCF factor between 0.02% and 5% is used to calculate the MTBF.
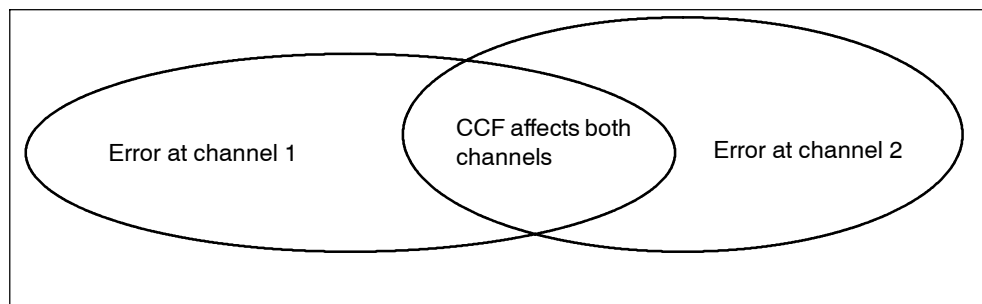
Figure A-3   Common Cause Failure (CCF)

Reliability of an S7-400H

The use of redundant modules prolongs the system MTBF by a very large factor. The integrated high–grade self–test and test / message functions of the S7-400H CPU enable the detection and localization of virtually all errors. The calculated diagnostics coverage lies by approx. 90%.

The reliability in stand–alone mode is described by the corresponding error rate. This corresponds with the reciprocal value of the MTTF (Mean Time To Failure, time between two errors). The MTTF is equivalent to the MTBF, assuming an infinite repair time MDT. The error rate of an S7-400H is calculated according to the SN29500 standard.

The reliability in redundant mode is described by the corresponding error rate. This corresponds with the reciprocal value of the MTTF. Those combinations of failed modules which cause a system failure form the minimum sections. The minimum sections are described individually by the Markov model.

Availability

Availability is the probability that a system is operable at a given point of time. This can be enhanced by means of redundancy, for example, by using redundant I/O modules or multiple sensors at the same 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 diagnostics display.

The availability of a system is expressed as a percentage. It is defined by the mean time between failure (MTBF) and the mean repair time MTTR (MDT). The availability of a two-channel (1-of-2) redundant system can be calculated from the following formula:

$$V = \frac{MTBF_{1v2}}{MTBF_{1v2} + MDT} 100\%$$
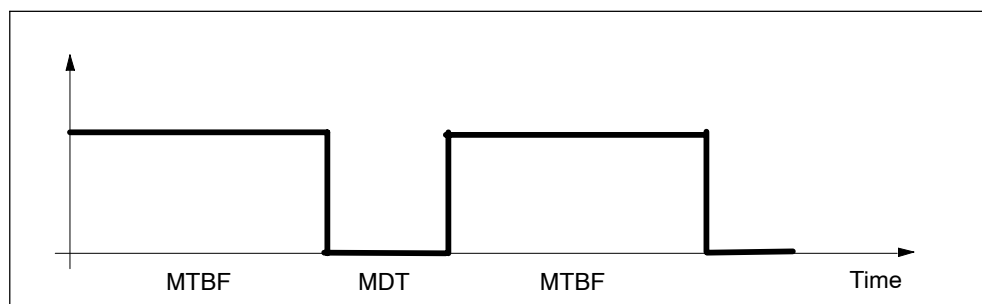
Figure A-4   Availability

A.2 Comparison of MTBFs for Selected Configurations

The following sections compare systems with a central I/O.
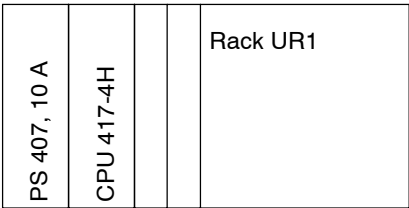
The following framework conditions are set for the calculation.

- MDT (Mean Down Time) 4 hours

- ambient temperature 40 degrees

- buffer voltage is guaranteed
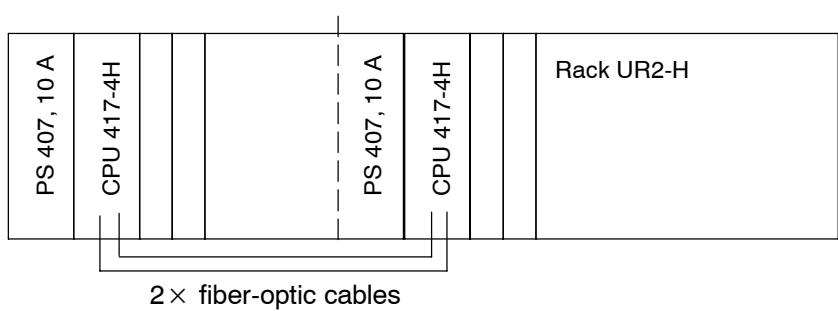
A.2.1 System configurations with central I/O
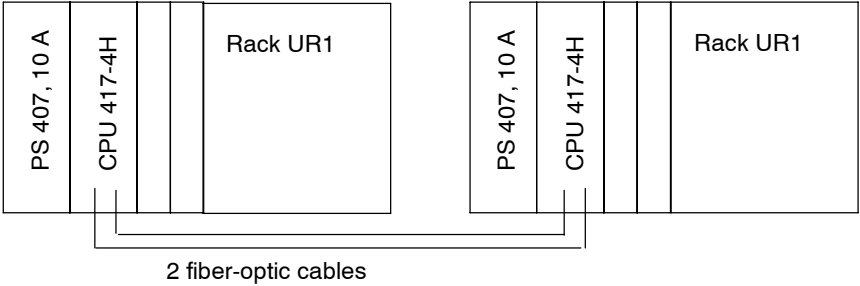
The following system containing one CPU (417-4H) CPU, for example) operating in stand–alone mode forms the basis for the calculation of a reference factor which defines the multiple of the availability of other systems with central I/O compared to the basic line.

Redundant CPU in stand–alone mode

Redundant CPU in stand–alone mode (417-4H CPU, for example)	Factor
PS 407, 10 A · CPU 417-4H · Rack UR1	1

Redundant CPUs in different racks
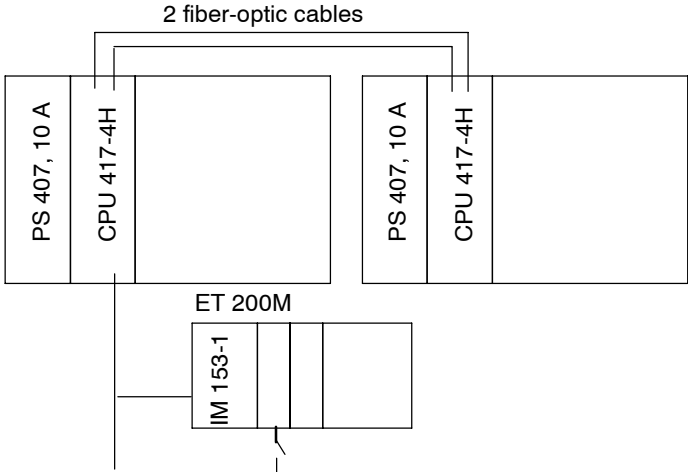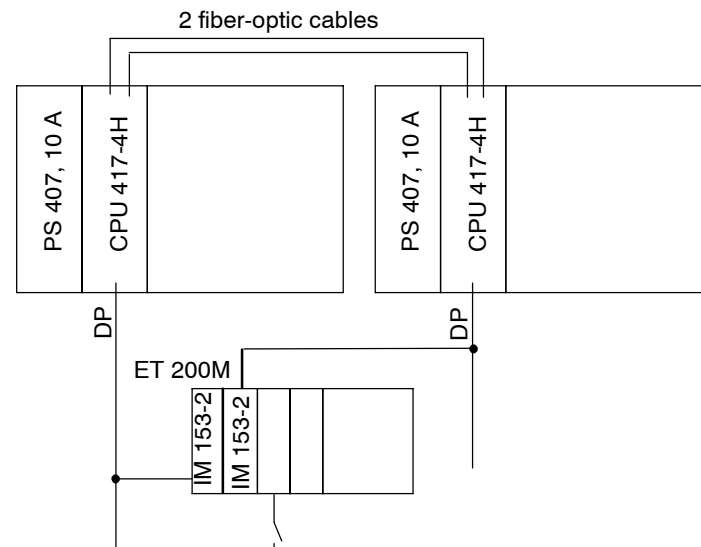
Redundant CPU 417-4H in a split rack, CCF = 2 %	Factor
PS 407, 10 A · CPU 417-4H · PS 407, 10 A · CPU 417-4H · Rack UR2-H 2× fiber-optic cables	20

Redundant CPU 417-4H in separate racks, CCF = 1 %	Factor
PS 407, 10 A │ CPU 417-4H │ │ │ Rack UR1 │ │ │ │ │ │ │ PS 407, 10 A │ CPU 417-4H │ │ │ Rack UR1 2 fiber-optic cables	38
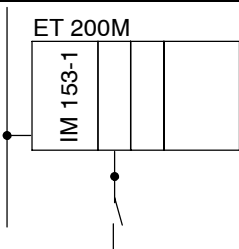
A.2.2   System configurations with distributed I/O

The system with two redundant CPUs 417-4 H and a one-sided I/O described below 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 basic line.

Redundant CPUs with single-channel, one-sided or switched I/O

One-sided distributed I/O	Base line
	1

Switched distributed I/O, CCF = 2 %	Factor
	15

Redundant CPUs with redundant I/O

Single-channel, one-sided I/O	MTBF factor
ET 200M IM 153-1	1
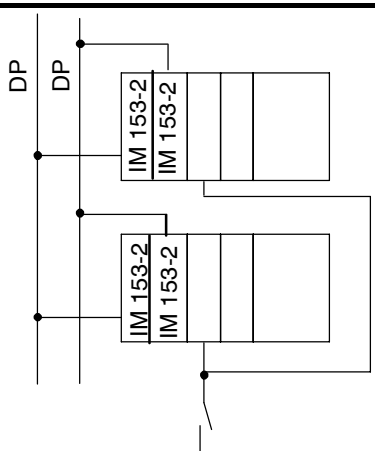
Redundant I/O	MTBF factor
DP DP IM 153-2 IM 153-2 IM 153-2 IM 153-2	**see table below**

Table A-1   MTBF factor for redundant I/O

Modules	MLFB	MTBF factor CCF = 1 %	MTBF factor CCF = 0.2 %
Digital input module, distributed			
DI 24xDC24V	6ES7 326-1BK00-0AB0	100	500
DI 8xNAMUR [EEx ib]	6ES7 326-1RF00-0AB0	100	500
DI16xDC24V, interrupt	6ES7 321-7BH00-0AB0	4	4
Analog input module, distributed			
AI 6x13Bit	6ES7 336-1HE00-0AB0	100	500
AI8x12Bit	6ES7 331-7KF02-0AB0	5	5
Digital output module, distributed			
DO 10xDC24V/2A	6ES7 326-2BF00-0AB0	100	500
DO8xDC24V/2A	6ES7 322-1BF01-0AA0	3	4
DO32xDC24V/0.5A	6ES7 322-1BL00-0AA0	3	4

Summary

There are now several thousand applications of redundant automation systems in the field, each with a different configuration. To calculate the MTBF, we assumed an average configuration.

Based on experience in the field, we may assume a total operating time of all redundant automation systems of 300 000 000 h. We have received reports of the failure of altogether four redundant automation systems.

This proves an assessed MTBF 95% over a period of 3000 years sufficiently reliable.

The MTBF values assessed as being real are:

Type I b, CCF = 2 %   approx. 230 a

Type I b, CCF = 0.2 %   approx. 1200 a
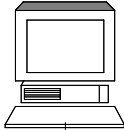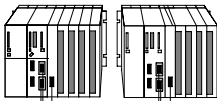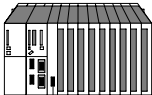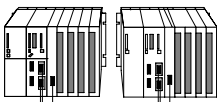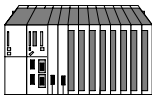
Type I differs compared to an average redundant automation system only in the use of a redundant power supply. Hence these considerations are rather pessimistic.

A.2.3 Comparison of system configurations with standard and redundant communication

The next section shows a comparison between standard and redundant communication for a configuration consisting of a redundant system, a redundant CPU operating in stand–alone mode, and of a single-channel OS.

In the comparison, we have only made allowances for the communication components CP and cable.

Systems with standard and redundant communications

Standard communication	Base line
OS workstation　　S7-400H system　　S7-400 with redundant CPU	1

Redundant communication	Factor
OS workstation　　S7-400H system　　S7-400 with redundant CPU	approx. 80

Stand−alone operation

B

Overview

This appendix offers information you to operate a redundant CPU in stand−alone mode (414-4H or 417-4H CPUs). You will learn:

- how stand−alone operation is defined
- when stand−alone operation is required
- what you have to take into account for stand−alone operation
- how the redundancy-specific LEDs react
- how to configure stand−alone operation of a redundant CPU
- how you can expand it to form a redundant system

The differences compared to a standard S7-400 CPU that you have to take into account when configuring and programming the redundant CPU are contained in Appendix D.

Definition

The term stand−alone operation refers to the use of a redundant CPU in a standard SIMATIC-400 station.

Reasons for stand−alone operation

The applications outlined below are only possible when using a redundant CPU, i.e. they are not operable on standard S7−400 CPUs.

- Using of redundant connections
- Configuration of the S7-400F fail-safe PLC

    A fail-safe user program can only be compiled for execution on a redundant CPU with a fail-safe F−Runtime license. For further information, refer to the *S7-400F and S7-400FH Programmable Controllers* manuals.

Note

The redundant CPU is also capable of performing the self-test when operating in stand−alone mode.

Allowances to be made for operating a redundant CPU in stand−alone mode

Notice

In the case of standalone operation of an H CPU, no synchronization modules may be connected. "0" must be set as the module carrier number.

Although a redundant CPU has additional functions compared to a standard S7-400 CPU, it does not support specific functions. In particular when you are programming your automation system, you should therefore know on which CPU you are going to run the use. A user program written for a standard S7-400 CPU usually will not run on a redundant CPU in stand−alone mode without further adaptations.

The table below lists the differences between the operation of a redundant CPU in stand−alone mode and in redundant mode.

Table B-1   Differences between standalone mode and redundant mode

Function	H-CPU in stand−alone mode	H-CPU redundantmode
Connection of S5 modules using an IM or adapter casings	via IM 463-2	No
Redundancy error OBs (OB70, OB72)	Yes, but no calls	Yes
CPU hardware error (OB 84)	after the detection and elimination of memory errors	after the detection and elimination of memory errors with reduced performance of the redundant coupling between both CPUs
SSL ID W#16#0232 index W#16#0004 byte 0 of the word "index" in the data record	W#16#F8	Stand−alone mode: W#16#F8 or W#16#F9 Redundant: W#16#F8 and W#16#F1 or W#16#F9 and W#16#F0
Multi-DP master mode	Yes	No
System modifications in run	Yes, as described in the "System Modification during Operation Using CIR" manual.	Yes, as described in chapter 11, redundant operation.

H–specific redundant LEDs

The REDF, IFM1F, IFM2F, MSTR, RACK0 and RACK1 LEDs show the reaction in stand–alone mode as specified in the table below.

LED	Reaction
REDF	Dark
IFM1F	Dark
IFM2F	Dark
MSTR	Lit
RACK0	Lit
RACK1	Dark

Configuring stand–alone mode

Prerequisite: the redundant CPU is not equipped with a synchronization module.

Procedure:

1. Insert a SIMATIC-**400** station in your project.

2. Configure the station with the redundant CPU according to your hardware setup. For stand–alone operation, insert the redundant CPU in a standard rack (Insert > Station > S7-400 Station in SIMATIC Manager).

3. Configure the parameters of the redundant CPU. Use the default values, or customize the necessary parameters.

4. Configure the necessary networks and connections. In your stand–alone configuration, you can implement "redundant S7 connections".

For help on procedures, refer to the help in SIMATIC Manager.

Expansion to a redundant system

Note

You can only expand your system for operation in redundant mode if you have not assigned any odd numbers to expansion units in stand−alone mode.

To expand the redundant CPU later to form a redundant system, proceed as follows:

1. Open a new project and insert a redundant station.

2. Copy the entire rack twice from the standard SIMATIC-400 station to the H station.

3. Insert the subnets as required.

4. Copy the DP slaves from the old stand−alone project to the redundant station as required.

5. Create a new configuration of the communication connections.

6. Carry out all changes required, such as the insertion of one-sided I/O.

For information on the project configuration, refer to the Online Help.

Changing the operating mode of a redundant CPU

The procedure in changing the operating mode of a redundant CPU is based on the current operating mode and rack number of the CPU:

Changing from redundant to stand−alone mode

1. Remove the synchronization modules.

2. Expand the CPU

3. Assign rack number 0 to the CPU.

4. Install the CPU

5. Download a project with the stand−alone configuration to the CPU.

Changing from stand−alone mode to redundant Mode, rack number 0

1. Insert the synchronization modules into the CPU.

2. Perform an unbuffered POWER ON, for example, by removing and inserting the CPU, or download the project containing the stand−alone configuration to the CPU.

Changing from stand−alone mode to redundant Mode, rack number 1

1. Assign rack number 1 to the CPU.

2. Install the CPU

3. Insert the synchronization modules into the CPU.

System modification in run in stand–alone mode

H–CPUs operating in stand–alone mode also support system modifications and certain configuration changes in RUN. The procedure corresponds with that for standard -CPUs. Processing is here halted, but not exceeding a duration of 2.5 seconds (configurable), and the process outputs retain their current values. In particular in process engineering systems, this has virtually no effect on the active process. See also the *"Modifications to the System During Operation Using CiR"* manual

System modifications in RUN are only supported with distributed I/O. Prerequisite is a configuration as shown in the figure below. For reasons of clarity, the view shows only one DP master system and only one PA master system.
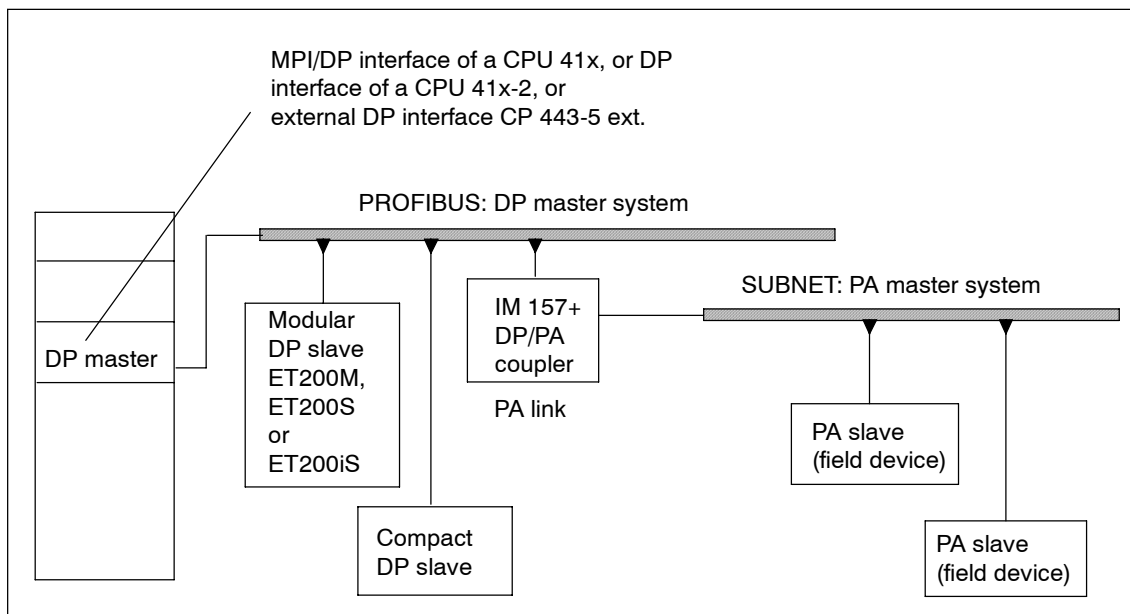
Figure B-1   Overview: system structure for configuration in run

Hardware requirements for system configuration in run

Hardware requirements for system configuration in run to be met in the commissioning phase:

- Use of an S7 400-CPU

- S7–400H CPU only in stand–alone mode

- Any CP 443-5 extended used must have a firmware V5.0 or later.

- To add modules to an ET 200M: installation of an IM153-2, version MLFB 6ES7 153-2BA00-0XB0 or later, or an IM153-2FO, version MLFB 6ES7 153-2BB00-0XB0 or later. The ET 200M installation also requires active backplane bus with sufficient free space for the planned expansion. The ET 200M installation must be compliant with IEC 61158.

- If you want to add entire stations: be sure to you have the required connectors, repeaters, etc.

- If you want to add PA slaves (field devices): use IM157 version MLFB 6ES7 157-0AA82-0XA00 or later in the corresponding DP/PA Link.

Note

You may generally mix components which support configuration in run with those that do not support this feature. Depending on your selected configuration, certain components may impose restrictions with respect to system configuration in run.

Software requirements for system configuration in run

You require STEP 7 V5.3. The user program must support system configuration in run in order to avoid a CPU STOP as a result of station failures or module errors.

Overview of permitted system modifications

You may modify the system configuration in run as described below:

- Adding components or modules with modular DP slaves ET 200M, ET 200S and ET 200iS, provided they are compliant with IEC 61158n

- The use of previously unused channels of a module of a module of the modular slaves ET 200M, ET 200S and ET 200iS

- Adding DP slaves to a DP master system.

- Adding PA slaves (field devices) to a PA master system

- Adding DP/PA couplers downstream of an IM157.

- Adding PA Links (including PA master systems) to a DP master system.

- Assigning added modules to a process image partition.

- Reconfiguration of I/O modules, for example, by setting different interrupt limits.

- Undoing changes: any components, modules, DP slaves and PA slaves (field devices) can be removed again.

Migrating from S5-H to S7-400H

C

This appendix will help you to convert to redundant S7 systems if you are already familiar with redundant 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 learning how to use the STEP 7 base software:

- *Configuring hardware and connections in STEP 7 V5.3*
- *Programming in STEP 7 V5.3*

Information on the various programming languages is found in the reference manuals listed below.

- *System and Standard Functions*
- *STL, LAD, FBD for S7-300/400*

The *From S5 to S7* manual supports you with details on migration.

C.2 Configuration, Programming and Diagnostics

Configuration

STEP 5 systems were configured using a separate configuration package, such as COM 155H.

In STEP 7, the redundant CPUs are configured using the standard software. In SIMATIC Manager, you can create a redundant station and configure it in HW CONFIG. The special features of the redundant CPUs are grouped in only a few tabs. The integration in networks and the configuration of connections are performed with NetPro.

Diagnostics and programming

In S5, error diagnostics are implemented with the help of the error data block to which the system writes all error data. Error OB 37 is started automatically when any entries are made. 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 in a bit pattern in the STEP 5 user program.

In STEP 7, system diagnostics is accomplished by means of the diagnostics buffer or by displaying what are known as partial lists from the system status list (specific information for redundant systems, for example, are located in SSL71). This check can be performed with the help of the PG or in the user program with SFC 51 "RDSYSST".

OB 70 and OB 72 are available for I/O redundancy loss and CPU redundancy loss.

The function of the control byte is implemented in STEP 7 by means of SFC 90 H_CTRL.

Topic in S5	Equivalent in S7
Error OB37	Error OBs OB 70 and OB 72
Memory control word	SFC 90 "H_CTRL"
Memory status word	SSL71
Error block	Diagnostics buffer

Differences Between Fault-Tolerant Systems and Standard Systems

D

When you configure and program a redundant automation system with redundant CPUs, make allowances for certain differences to the standard S7-400 CPUs. Although a redundant CPU has additional functions compared to a standard S7-400 CPU, it does not support specific functions. This has to be taken in account particularly if you wish to run a program that was created for a standard S7-400 CPU on a redundant CPU.

The items in which the programming of redundant systems differs from that for standard systems are summarized below. You will find further differences in Appendix B.

When using any of the relevant calls (OBs and SFCs) in your user program, adapt your program accordingly.

Additional functions of redundant systems

Function	Additional programming
Redundancy error OBs	• I/O redundancy error OB (OB 70) • CPU redundancy error OB (OB 72) For detailed information, refer to the *System and Standard Functions* reference manual.
CPU hardware error	OB84 is also called if the performance of the redundant coupling between the CPUs is reduced.
Additional information in OB start information and in diagnostics buffer entries	The rack number and the CPU (master/standby) are specified. You can evaluate this additional information in the program.
SFC for redundant systems	You can control processes in redundant systems using SFC 90 "H_CTRL".
Redundant communication connections	Redundant connections are configured and do not require further programming. You can use the SFBs for configured connections when using redundant connections.
Self-test	The self-test is performed automatically, no further programming is required,
High–quality RAM-Test	The CPU performs a high–quality RAM test after an unbuffered POWER ON.
Switched I/O	No additional programming required, see chapter 8.3.

Function	Additional programming
Information in the system status list	• You also obtain data records for the H–specific LEDs using the partial list using the SSL ID W#16#0019. • You also obtain data records for the redundancy error OBs using the partial list using the SSL ID W#16#0222. • You obtain information on the current state of the redundant system using the partial list with SSL ID W#16#xy71. • You also obtain data records for the H-specific LEDs using the partial list with the SSL ID W#16#0174. • The partial list with the SSL-ID W#16#xy75 provides information on the status of the communications between the redundant system and switched DP slaves.
Update monitoring	The operating system monitors the following four configurable timers: • maximum cycle time extension • Maximum communication delay • Maximum retention time for priority classes > 15 • Minimum I/O retention time No additional programming is required for this. For details, refer to chapter 7.
SSL ID W#16#0232 index W#16#0004 byte 0 of the word "index" in the data record	H–CPU in standalone mode: W#16#F8 H–CPU in 1–of–1 mode: W#16#F8 or W#16#F9 H-CPU in redundant mode: W#16#F8 and W#16#F1 or W#16#F9 and W#16#F0

Restrictions of the redundant CPU compared to a standard CPU

Function	Restriction of the redundant CPU
Hot restart	A hot restart is not supported. OB 101 is not supported
Multicomputing	Multicomputing is not supported. OB 60 and SFC 35 are not supported
Startup without configuration loaded	Startup without loaded configuration is not supported.
Background OB	OB 90 is not supported.
Multi-DP master mode	The H-CPUs do not support multi-DP master mode in the REDUNDANT operating mode.
Direct communication between DP slaves	Can not be configured in STEP 7
Constant bus cycle times of DP slaves	No constant bus cycle times for DP slaves in the redundant system
Synchronizing DP slaves	Synchronization of DP slave groups is not supported. SFC 11 "DPSYC_FR" is not supported.
Disabling and enabling DP slaves	Disabling and enabling DP slaves is not supported. SFC 12 "D_ACT_DP" is not supported.

Function	Restriction of the redundant CPU
Insertion of DP modules in the module slots for interface modules	Not supported The module slots are designed only for use by synchronization modules.
Runtime response	The command execution time of 414-4H and 417-4H CPUs is slightly longer than that of the corresponding standard CPU (414–4 and 417-4 CPUs) (see *S7-400 Operations List*). Make allowances for this for all time-sensitive applications. You may need to increase the cycle monitoring time.
DP cycle time	414-4H and 417-4H CPUs have a slightly longer cycle time than the corresponding standard CPU.
Delays and inhibits	During update: • the system outputs a negative acknowledgement for asynchronous SFCs for data records. • Messages are delayed • All priority classes up to 15 are initially delayed • Communications requests are rejected or delayed • Finally, all priority classes are inhibited. For more details, refer to chapter 7.
Use of symbol-oriented messages (SCAN)	The use of symbol-oriented messages is not supported.
Global data communication	GD communication is not supported (neither cyclically, nor by calling system functions SFC 60 "GD_SND" and SFC 61 "GD_RCV")
S7 basic communication	Communication functions (system functions) for basic communication are not supported.
S5 connection	The connection S5 modules by means of adapter module is not supported. The connection of S5 modules via IM 463-2 is only supported in stand–alone mode.
CPU as DP Slave	Not supported
Using SFC49 "LGC_GADR"	You operate an S7-400H automation system in redundant mode. If you declare the logical address of module of the switched DP slave at the LADDR parameter and call SFC49, the high byte of the RACK parameter returns the DP master system ID of the active channel. If there is no active channel, the function outputs the ID of the DP master system belonging to the master CPU.
Call of SFC51 "RDSYSST" with SSL_ID=W#16#xy91	The data records of the SSL partial lists shown below can not be read with SFC51 "RDSYSST": • SSL_ID=W#16#0091 • SSL_ID=W#16#0191 • SSL_ID=W#16#0291 • SSL_ID=W#16#0391 • SSL_ID=W#16#0991 • SSL_ID=W#16#0

Function modules and communication processors supported by the S7-400H

E

You can use the following function modules (FMs) and communication processors (CPs) on an S7-400:

FMs and CPs used centrally

Module	Order no.	Release	one-sided	redundant
Counter module FM 450	6ES7 450-1AP00-0AE0	Version 2 or later	Yes	No
Communication module CP441-1 (PtP connection)	6ES7 441-1AA02-0AE0	Version 2 or later	Yes	No
	6ES7 441-1AA03-0XE0	Version 1 or later with firmware V1.0.0		
Communication module CP441-2 (PtP connection)	6ES7 441-2AA02-0AE0	Version 2 or later	Yes	No
	6ES7 441-2AA03-0XE0	Version 1 or later with firmware V1.0.0		
Communications processor CP443-1 Multi (SINEC H1 (Ethernet), TCP/ISO transport)	6GK7 443-1EX10-0XE0	Version 1 or later with firmware V1.0.1	Yes	Yes
	6GK7 443-1EX11-0XE0	Version 1 or later with firmware V1.1.0	Yes	Yes
Communications processor CP443-5 Basic (PROFIBUS; S7 communications)	6GK7 443-5FX01-0XE0	Version 1 or later with firmware V3.1	Yes	Yes
Communication processor CP443-5 Extended (PROFIBUS; master on PROFIBUS-DP) [1]	6GK7 443-5DX02-0XE0	Version 2 or later with firmware V3.2.3	Yes	Yes
Communication processor CP443-5 Extended (PROFIBUS DPV1) [1] [2]	6GK7 443-5DX03-0XE0	Version 2 or later with firmware V5.0.0	Yes	Yes

[1] Only these modules should be used as external master interfaces on the PROFIBUS DP.

[2] Only this module supports DPV1 as external DP master interface (in accordance with IEC 61158/ EN 50170).

FMs and CPs for distributed one-sided use

> **Note**
>
> You can use all the FMs and CPs released for the ET 200M with the S7-400H distributed and one-sided.

FMs and CPs for distributed switched use

Modules	Order no.	Release
Communication processor CP 341-1 (point-to-point connection)	6ES7 341-1AH00-0AE0 6ES7 341-1BH00-0AE0 6ES7 341-1CH00-0AE0	Version 3 or later
	6ES7 341-1AH01-0AE0 6ES7 341-1BH01-0AE0 6ES7 341-1CH01-0AE0	Version 1 or later with firmware V1.0.0
Communication processor CP 342-2 (ASI bus interface module)	6GK7 342-2AH01-0XA0	Version 1 or later with firmware V1.10
Communication processor CP 343-2 (ASI bus interface module)	6GK7 343-2AH00-0XA0	Version 2 or later with firmware V2.03
Counter module 350-1	6ES7 350-1AH01-0AE0 6ES7 350-1AH02-0AE0	Version 1 or later
Counter module 350-2	6ES7 350-2AH00-0AE0	Version 2 or later
Controller module FM 355 C	6ES7 355-0VH10-0AE0	Version 4 or later
Controller module FM 355 S	6ES7 355-1VH10-0AE0	Version 3 or later
High Speed Boolean Processor FM 352-5	6ES7352-5AH00-0AE0	Version 1 or later with firmware V1.0.0
Controller module FM 355–2 C	6ES7 355-0CH00-0AE0	Version 1 or later with firmware V1.0.0
Controller module FM 355–2 S	6ES7 355-0SH00-0AE0	Version 1 or later with firmware V1.0.0

> **Notice**
>
> One-sided or switched function and communication modules will **not** be synchronized in the system if these exist in pairs. Example: two identical FM 450 modules operating in one-sided mode do **not** synchronize their counter states.

Redundant PROFIBUS DP slaves

STEP 7 V 5.3 with HW update is prerequisite for the operation of redundant PROFIBUS DP slaves.

Connection Examples for Redundant I/O

F

F.1 SM 321; DI 16 x DC 24 V, 6ES7 321-1BH02-0AA0

The diagram below shows the connection of two redundant encoders to two SM 321; DI 16 x DC 24 V. The encoders are connected to channel 0.
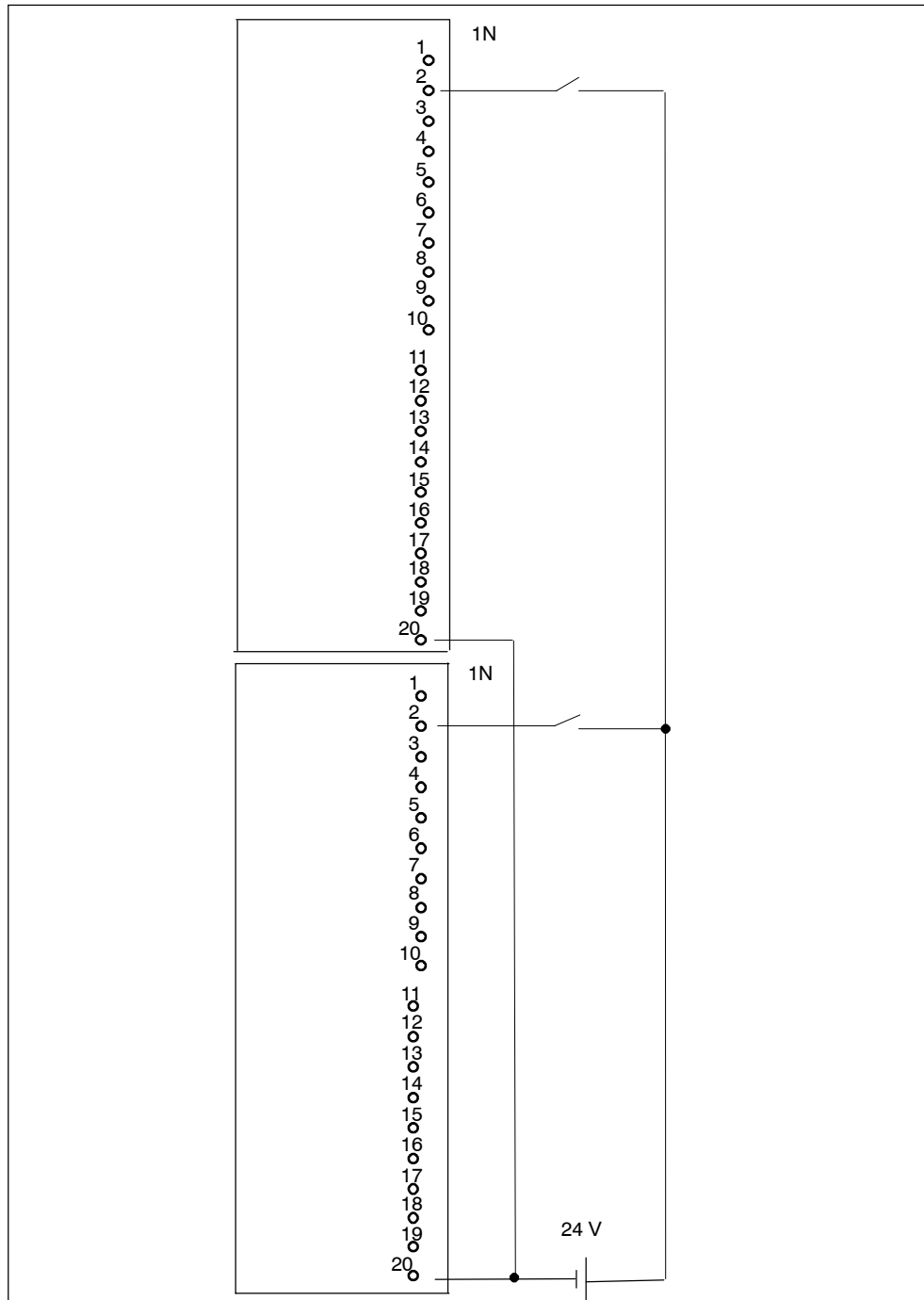
Figure F-1   Example of an SM 321 interconnection; DI 16 x DC 24 V

F.2 SM 321; DI 32 x DC 24 V, 6ES7 321-1BL00-0AA0

The diagram below shows the connection of two redundant encoder pairs to two redundant SM 32; DI 32 x DC 24 V. The encoders are connected to channel 0 and channel 16 respectively.
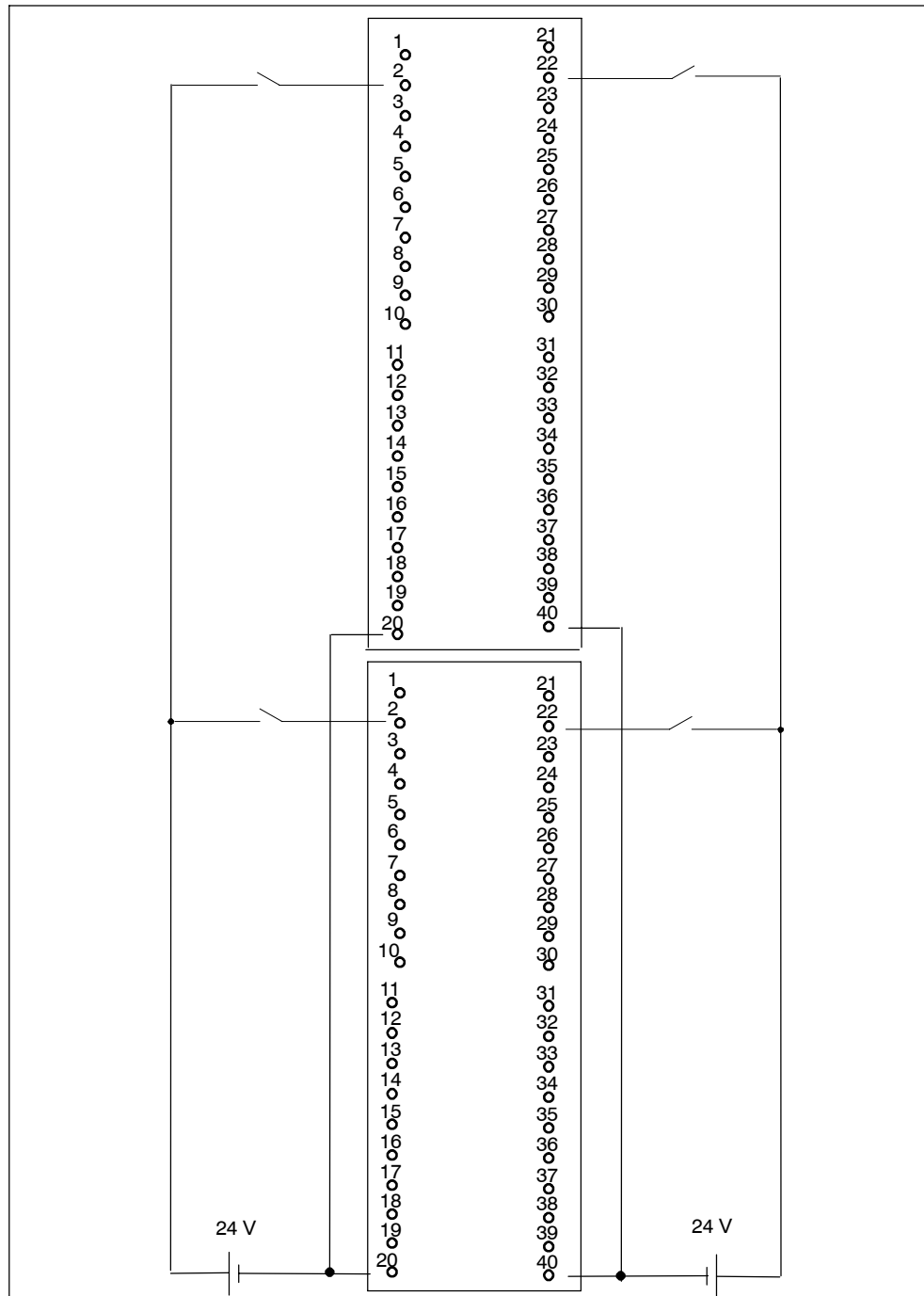
Figure F-2   Example of an SM 321 interconnection; DI 32 x DC 24 V

F.3   SM 321; DI 16 x AC 120/230V, 6ES7 321-1FF00-0AA0

The diagram below shows the connection of two redundant encoders to two
SM 321; DI 16 x AC 120/230 V. The encoders are connected to channel 0.
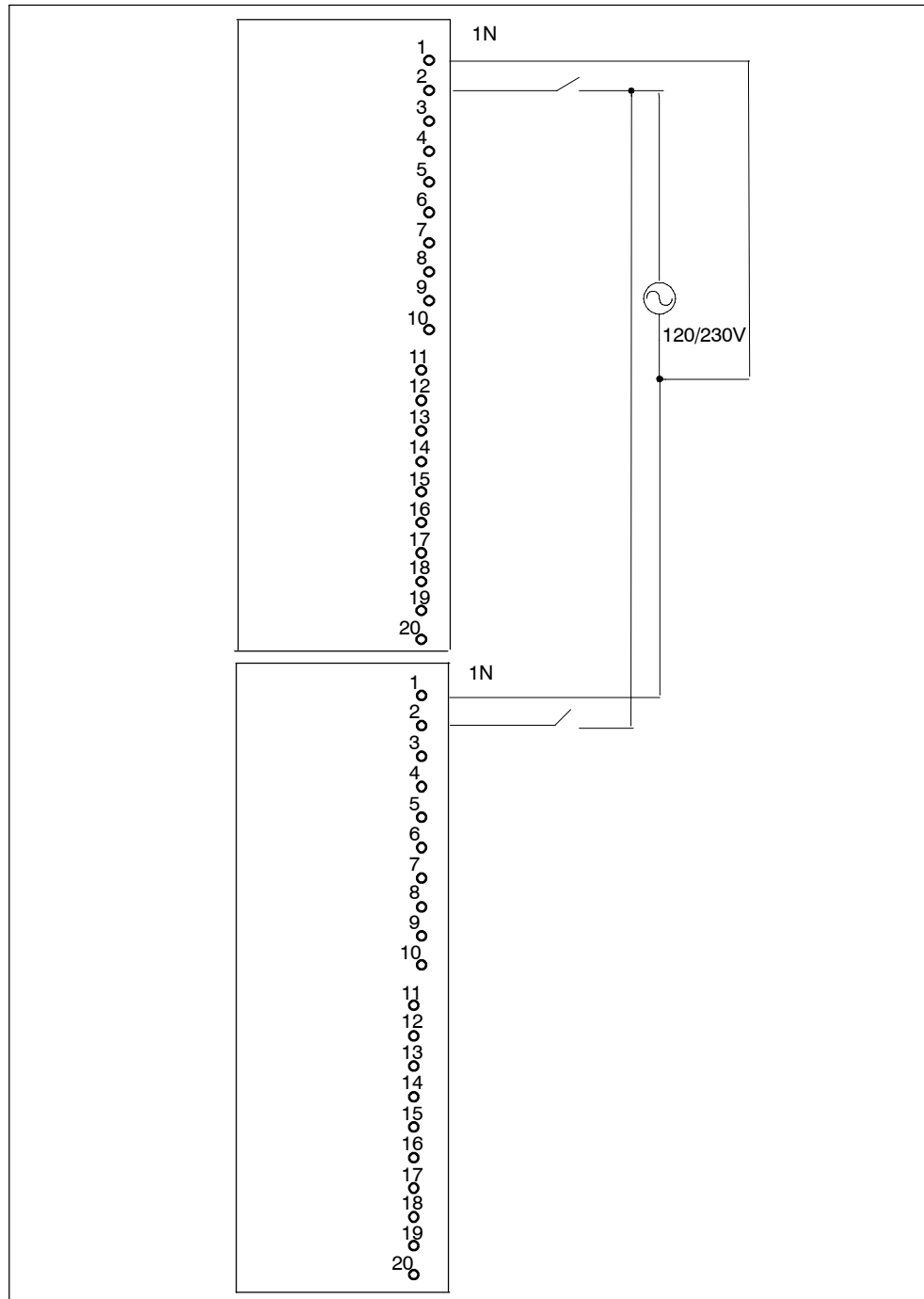
Figure F-3   Example of an interconnection with SM 321; DI 16 x AC 120/230 V

F.4 SM 321; DI 8 x AC 120/230 V, 6ES7 321-1FF01-0AA0

The diagram below shows the connection of two redundant encoders to two SM 321; DI 8 $\times$ AC 120/230 V. The encoders are connected to channel 0.
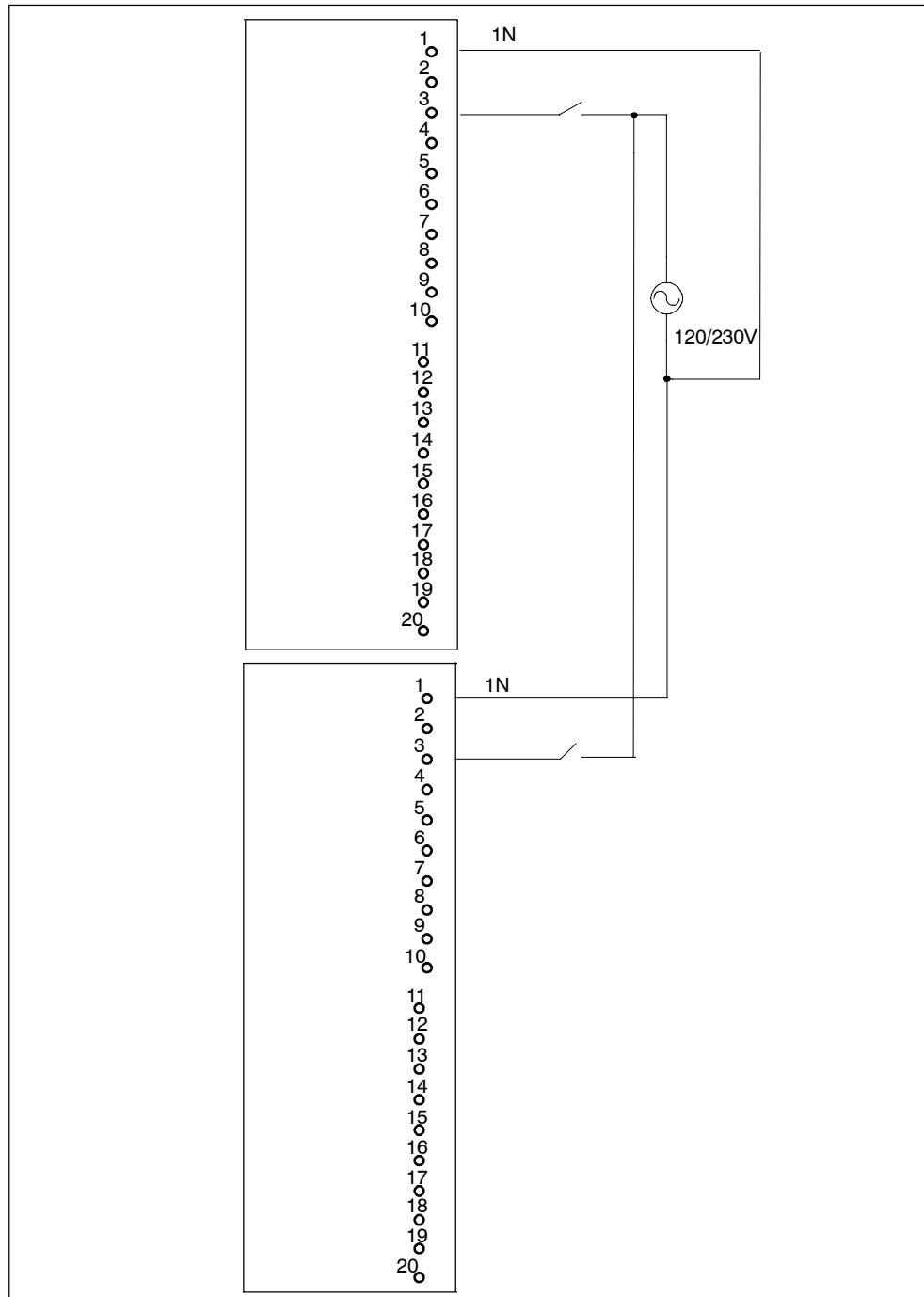
Figure F-4   Example of an interconnection with SM 321; DI 8 x AC 120/230 V

F.5 SM 321; DI 16 x DC 24V, 6ES7 321-7BH00-0AB0

The diagram below shows the connection of two redundant encoder pairs to two SM 321; DI 16 x DC 24 V. The encoders are connected to channel 0 or 8.
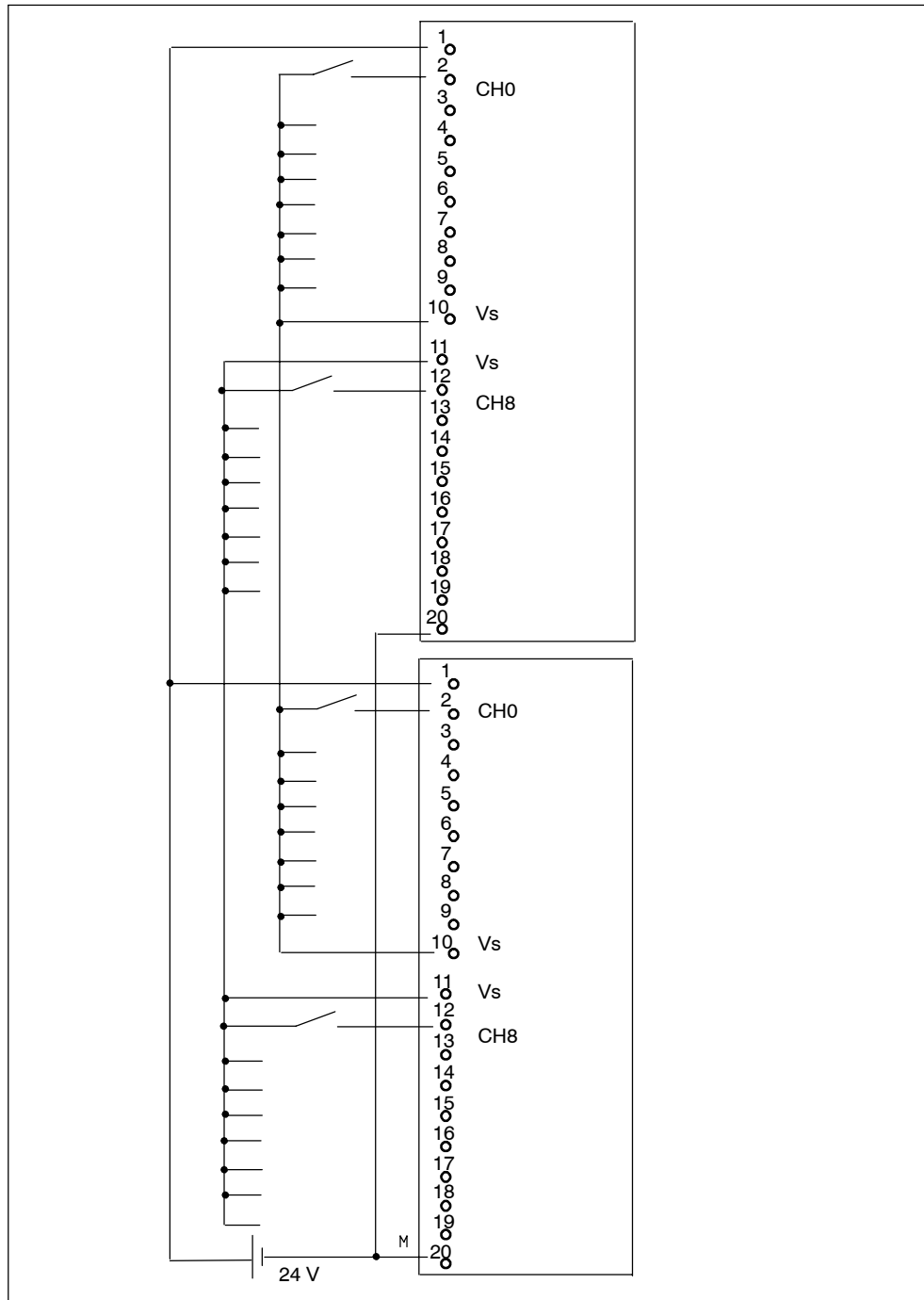
Figure F-5   Example of an interconnection with SM 321; DI 16 x DC 24V

F.6 SM 321; DI 16 x DC 24V, 6ES7 321-7BH01-0AB0

The diagram below shows the connection of two redundant encoder pairs to two SM 321; DI 16 x DC 24 V. The encoders are connected to channel 0 or 8.
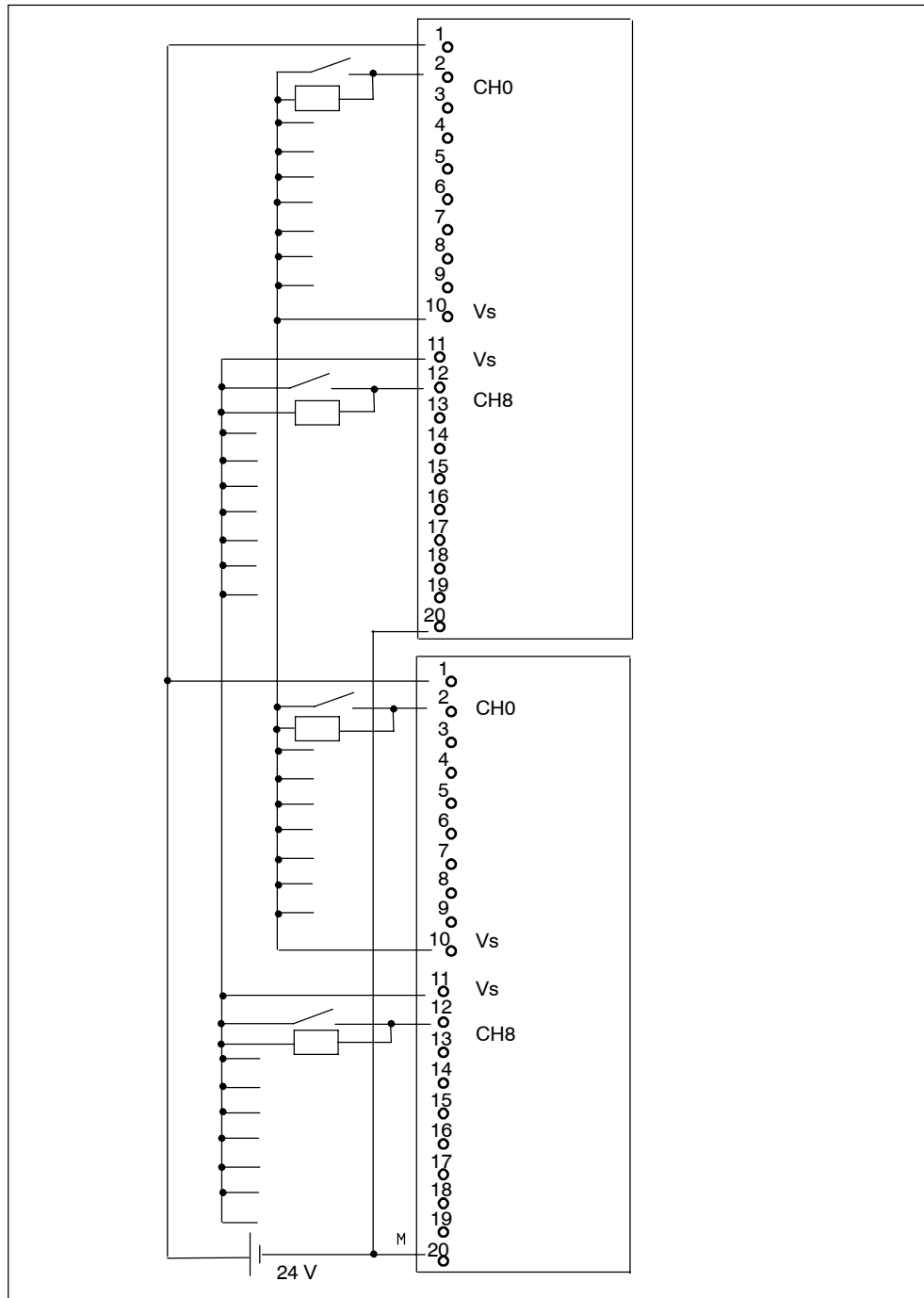
Figure F-6   Example of an interconnection with SM 321; DI 16 x DC 24V

F.7 SM 326; DO 10 x DC 24V/2A, 6ES7 326-2BF00-0AB0

The diagram below shows the connection of an actuator to two redundant SM 326; DO 10 x DC 24V/2AV. The actuator is connected to channel 1 respectively.
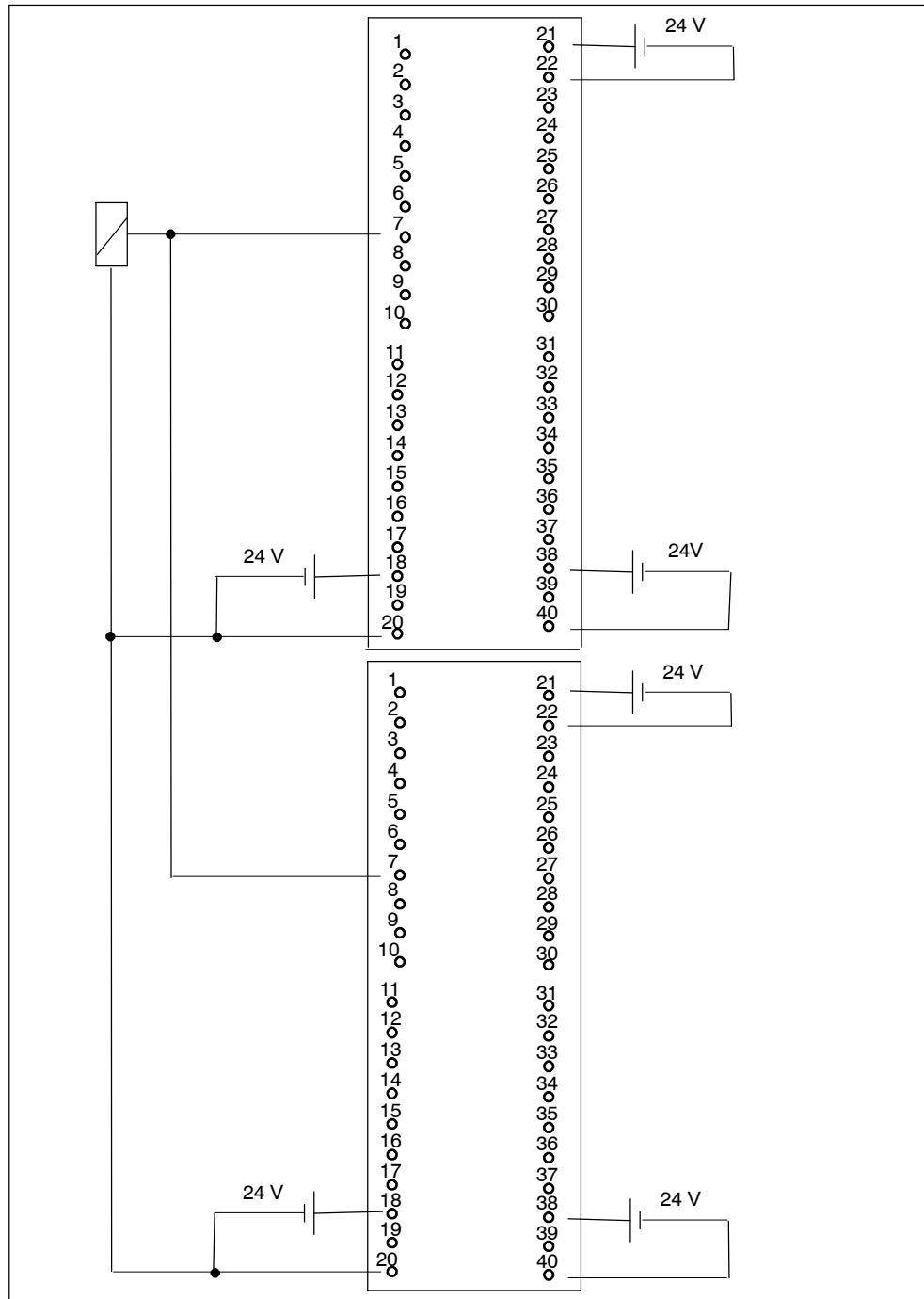
Figure F-7   Example of an interconnection with SM 326; DO 10 x DC 24 V/2 A

F.8 SM 326; DI 8 x NAMUR, 6ES7 326-1RF00-0AB0

The diagram below shows the connection of two redundant encoders to two redundant SM 326; DI 8 xNAMUR . The encoders are connected to channel 13.
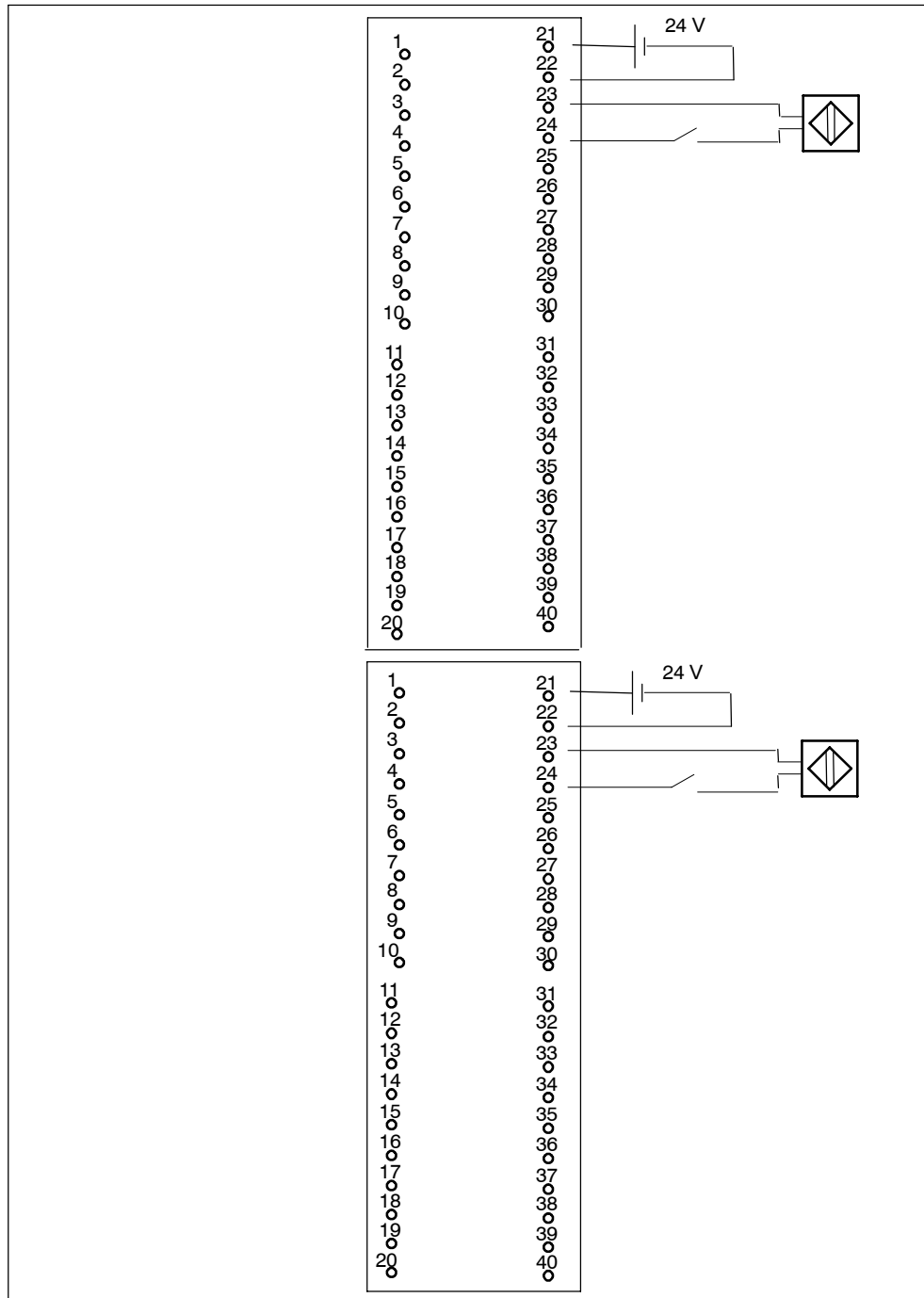
Figure F-8   Example of an interconnection with SM 326; DI 8 x NAMUR

F.9   SM 326; DI 24 x DC 24 V, 6ES7 326-1BK00-0AB0

The diagram below shows the connection of one encoder to two redundant
SM 326; DI 24 x DC 24 V. The encoder is connected to channel 13.
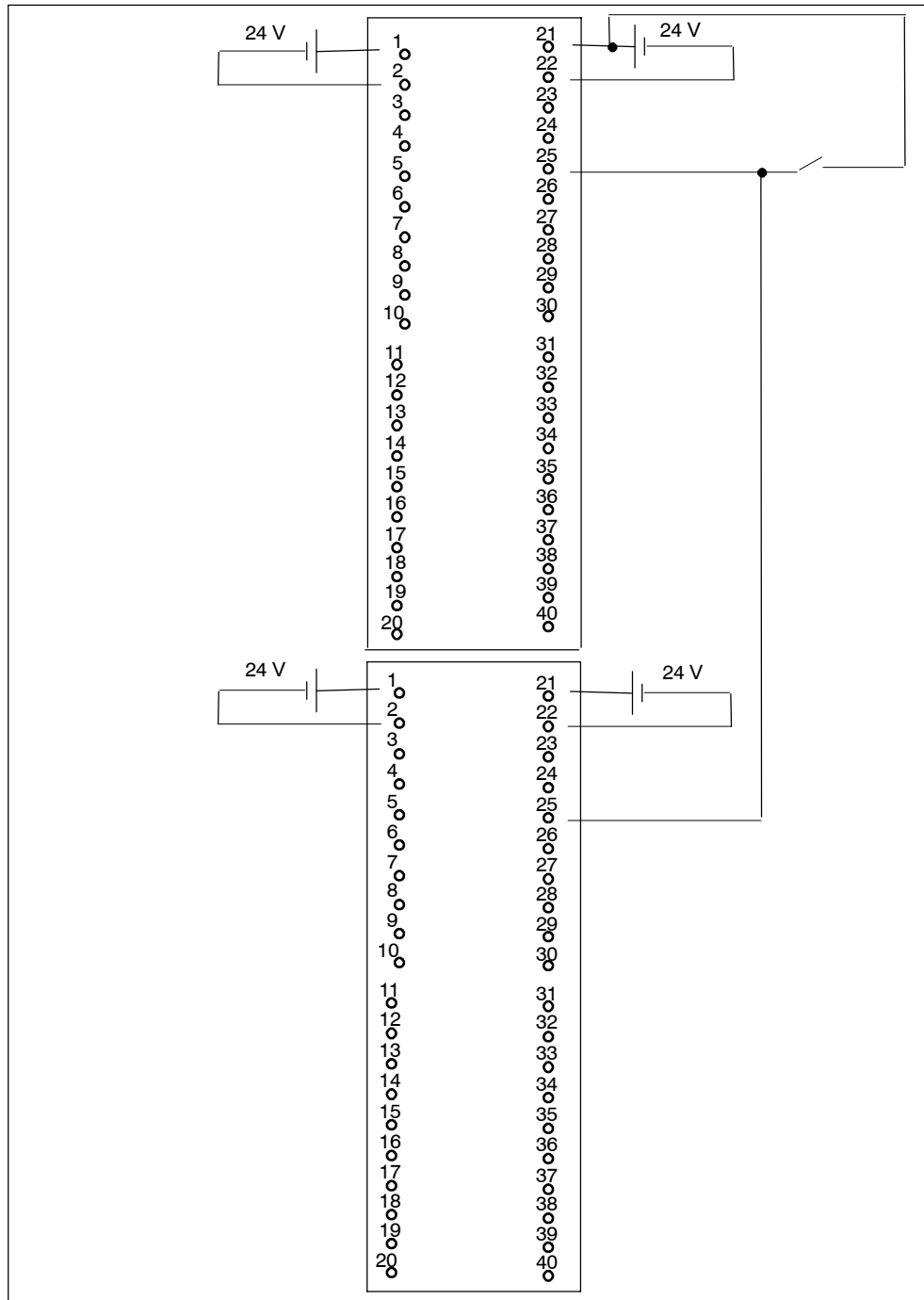
Figure F-9   Example of an interconnection with SM 326; DI 24 x DC 24 V

F.10   SM 421; DI 32 x UC 120 V, 6ES7 421-1EL00-0AA0

The diagram below shows the connection of a redundant encoder to two SM 421; DI 32 x UC 120 V. The encoder is connected to channel 0.
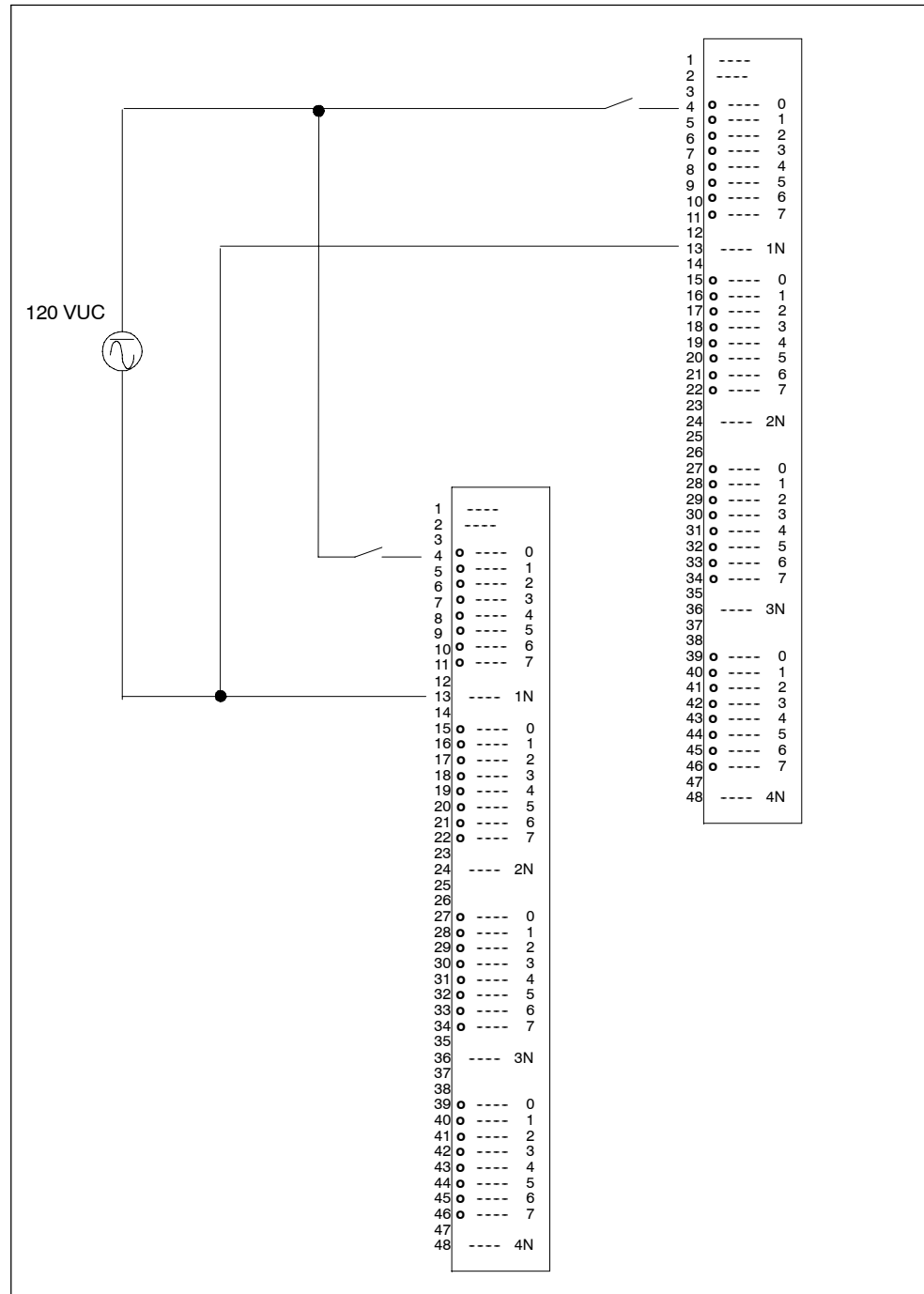
Figure F-10   Example of an interconnection with SM 421; DI 32 x UC 120 V

F.11   SM 421; DI 16 x DC 24 V, 6ES7 421-7BH01-0AB0

The diagram below shows the connection of two redundant encoders pairs to twoSM 421; D1 16 x 24 V. The encoders are connected to channel 0 and 8.
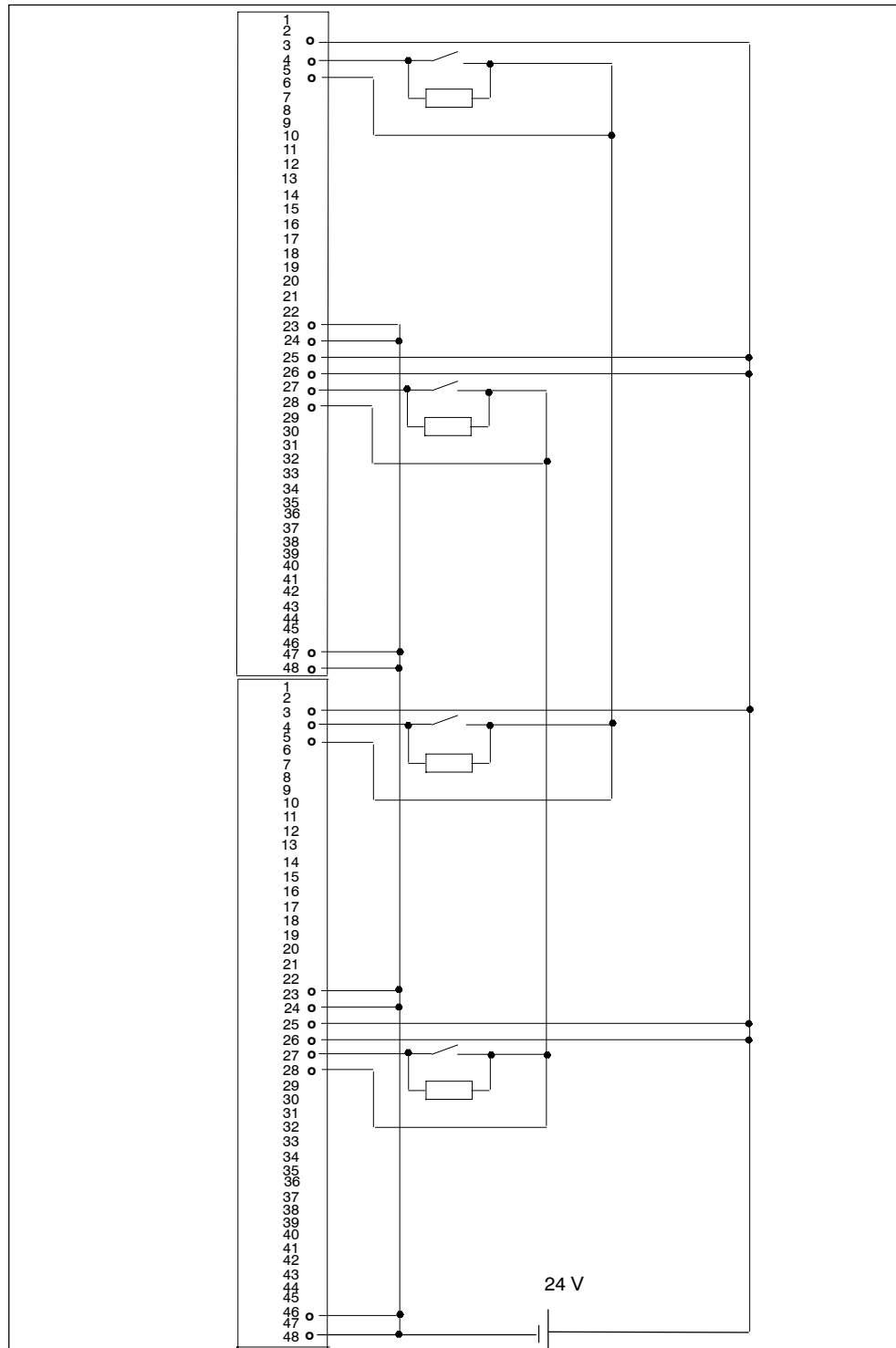
Figure F-11   Example of an interconnection with SM 421; DI 16 x 24 V

F.12 SM 421; DI 32 x DC 24 V, 6ES7 421-1BL00-0AB0

The diagram below shows the connection of two redundant encoders to two SM 421; D1 32 x 24 V. The encoders are connected to channel 0.
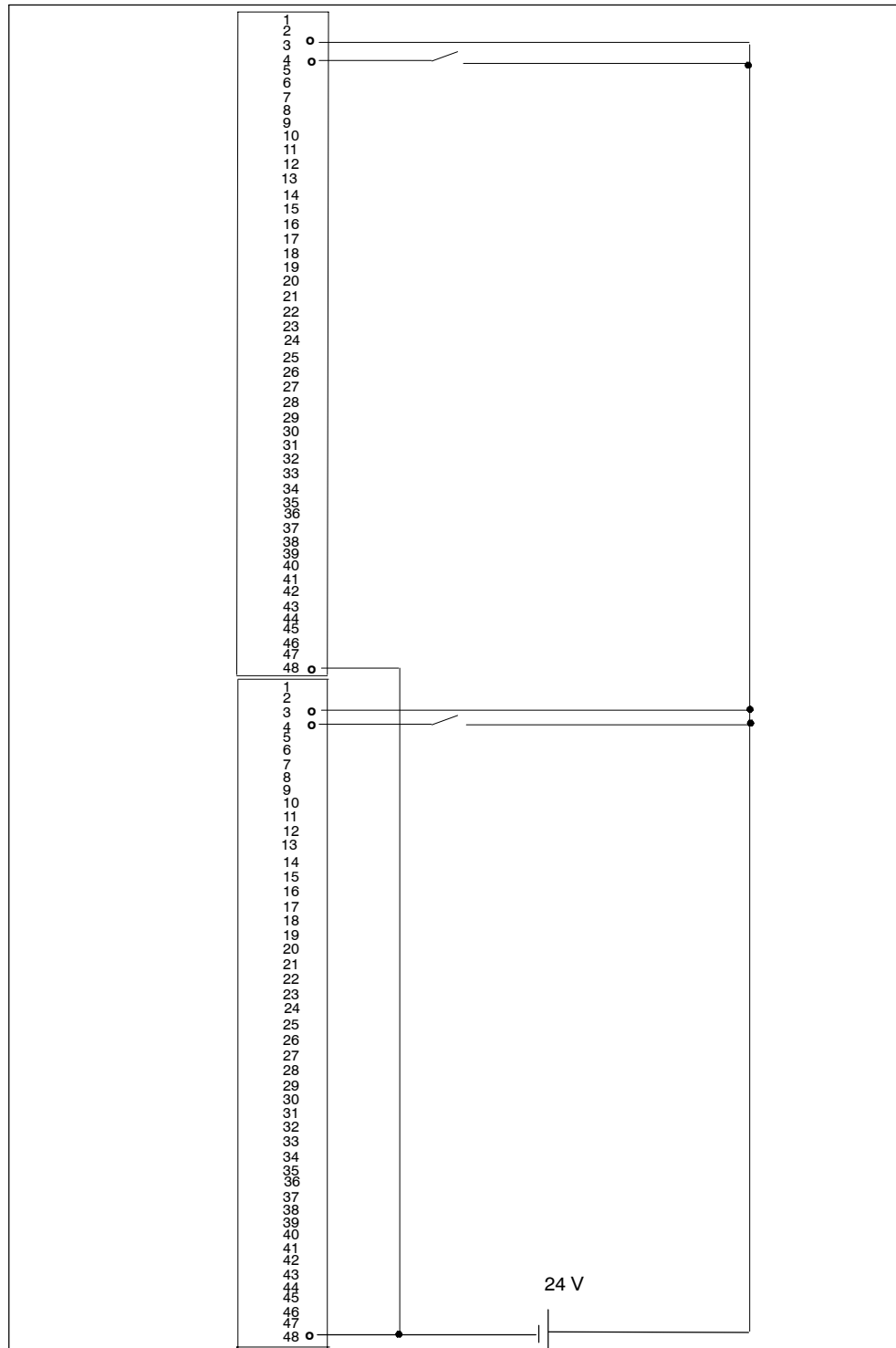
Figure F-12   Example of an interconnection with SM 421; DI 32 x 24 V

F.13 SM 421; DI 32 x DC 24 V, 6ES7 421-1BL01-0AB0

The diagram below shows the connection of two redundant encoders to two SM 421; D1 32 x 24 V. The encoders are connected to channel 0.
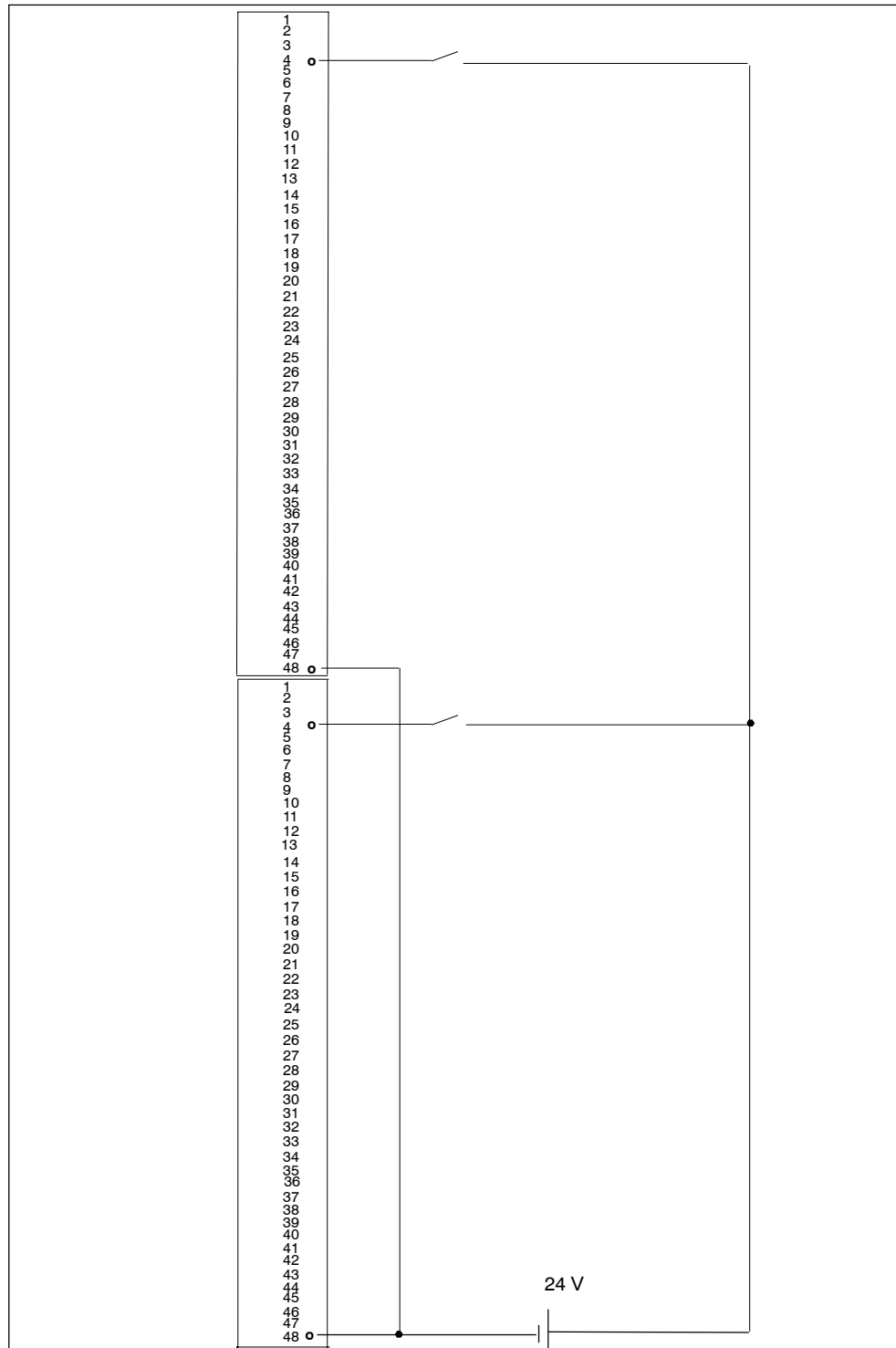
Figure F-13   Example of an interconnection with SM 421; DI 32 x 24 V

F.14　SM 322; DO 8 x DC 24V/2A, 6ES7 322-1BF01-0AA0

The diagram below shows the connection of an actuator to two redundant SM 322; DO 8 x DC 24 V. The actuator is connected to channel 0.

Suitable diodes are those of the series 1N4003 ... 1N4007, or any other diode with $V_r >= 200$ V and $I_F >= 1$ A
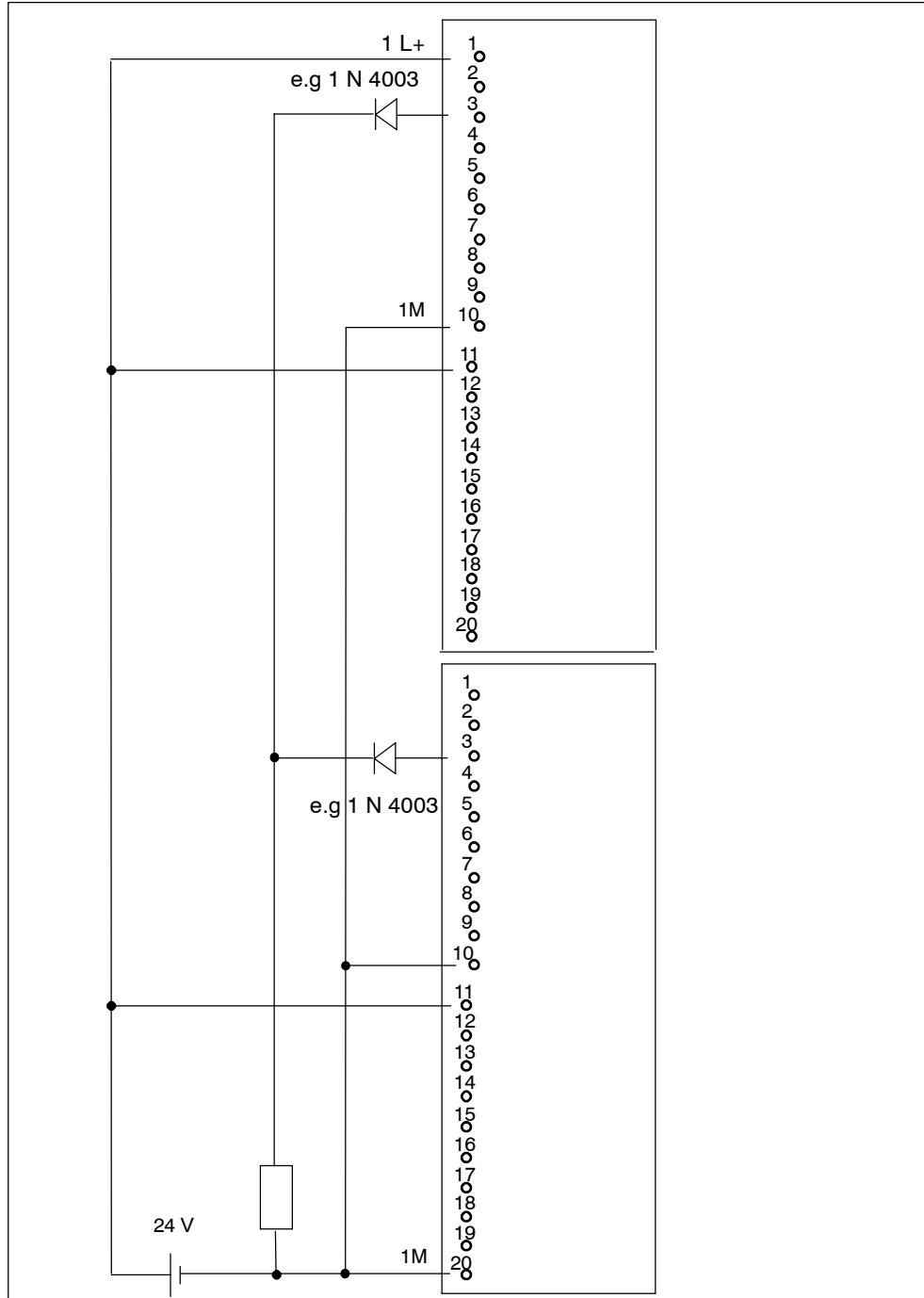
Figure F-14　Example of an interconnection with SM 322; DO 8 x DC 24 V/2 A

F.15 SM 322; DO 32 x DC 24 V/0.5 A, 6ES7 322-1BL00-0AA0

The diagram below shows the connection of an actuator to two redundant SM 322; DO 32 x DC 24 V. The actuator is connected to channel 1.

Suitable diodes are those of the series 1N4003 ... 1N4007, or any other diode with $V_r >= 200$ V and $I_F >= 1$ A
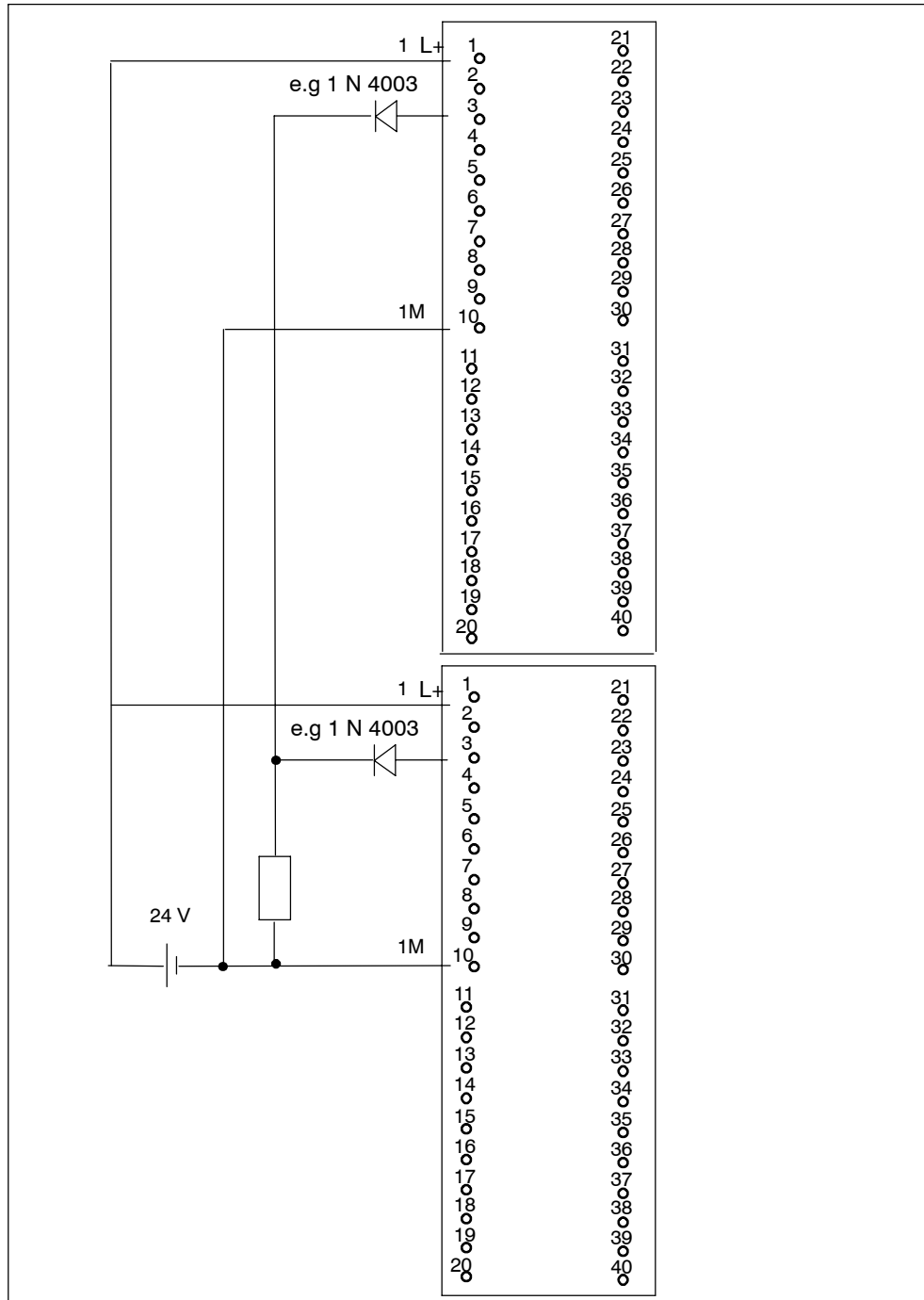
Figure F-15   Example of an interconnection with SM 322; DO 32 x DC 24 V/0.5 A

F.16 SM 322; DO 8 x AC 230 V/2 A, 6ES7 322-1FF01-0AA0

The diagram below shows the connection of an actuator to two SM 322;
DO 8 x AC 230V/2AV. The actuator is connected to channel 0.
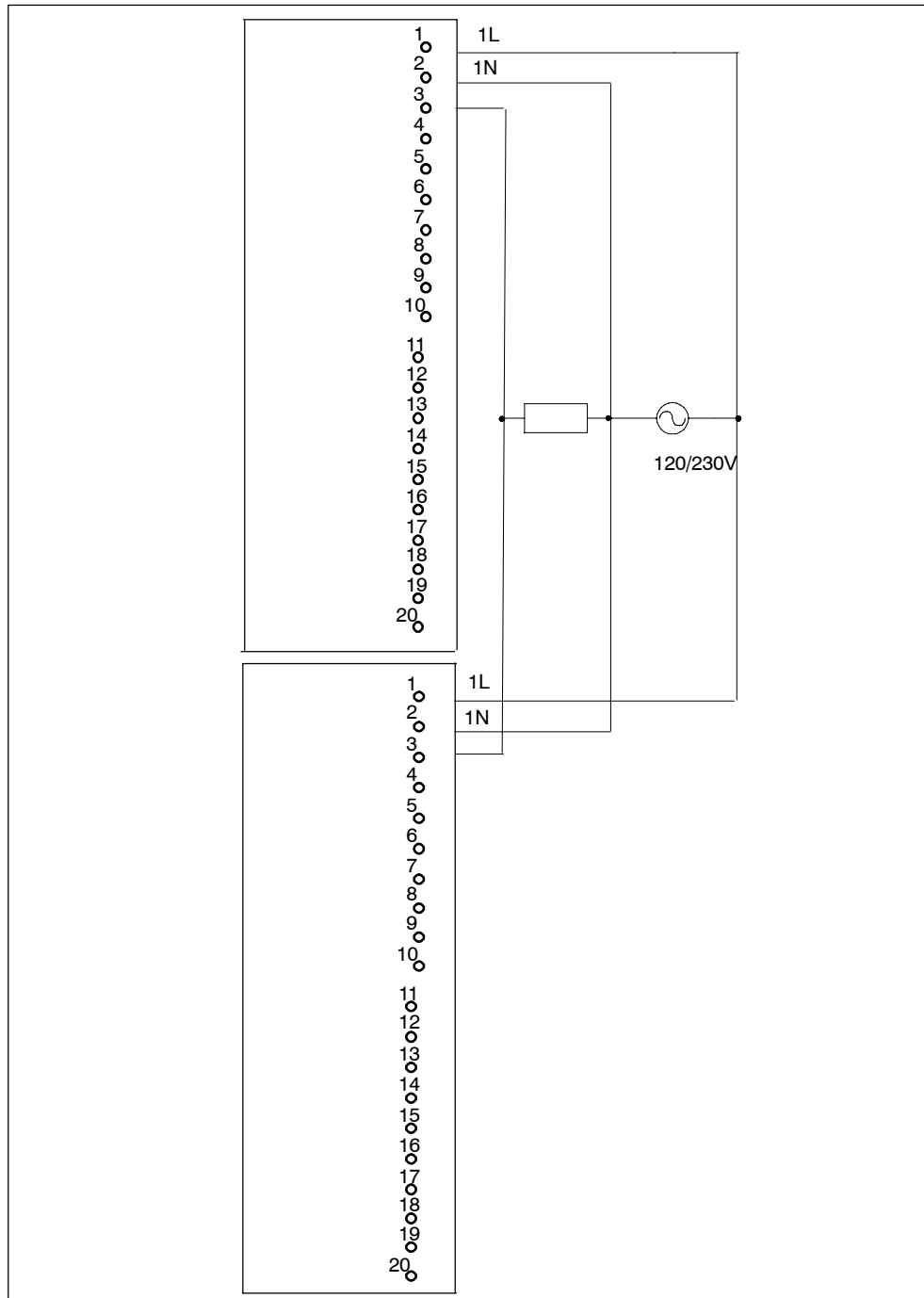
Figure F-16   Example of an interconnection with SM 322; DO 8 x AC 230 V/2 A

F.17 SM 322; DO 16 x DC 24 V/10 mA [EEx ib], 6ES7 322-5SD00-0AB0

The diagram below shows the connection of an actuator to two SM 322;
DO 16 x DC 24 V/10 mA [EEx ib]. The actuator is connected to channel 0.
Suitable diodes are those of the series 1N4003 ... 1N4007, or any other diode with
$V_r >= 200$ V and $I_F >= 1$ A
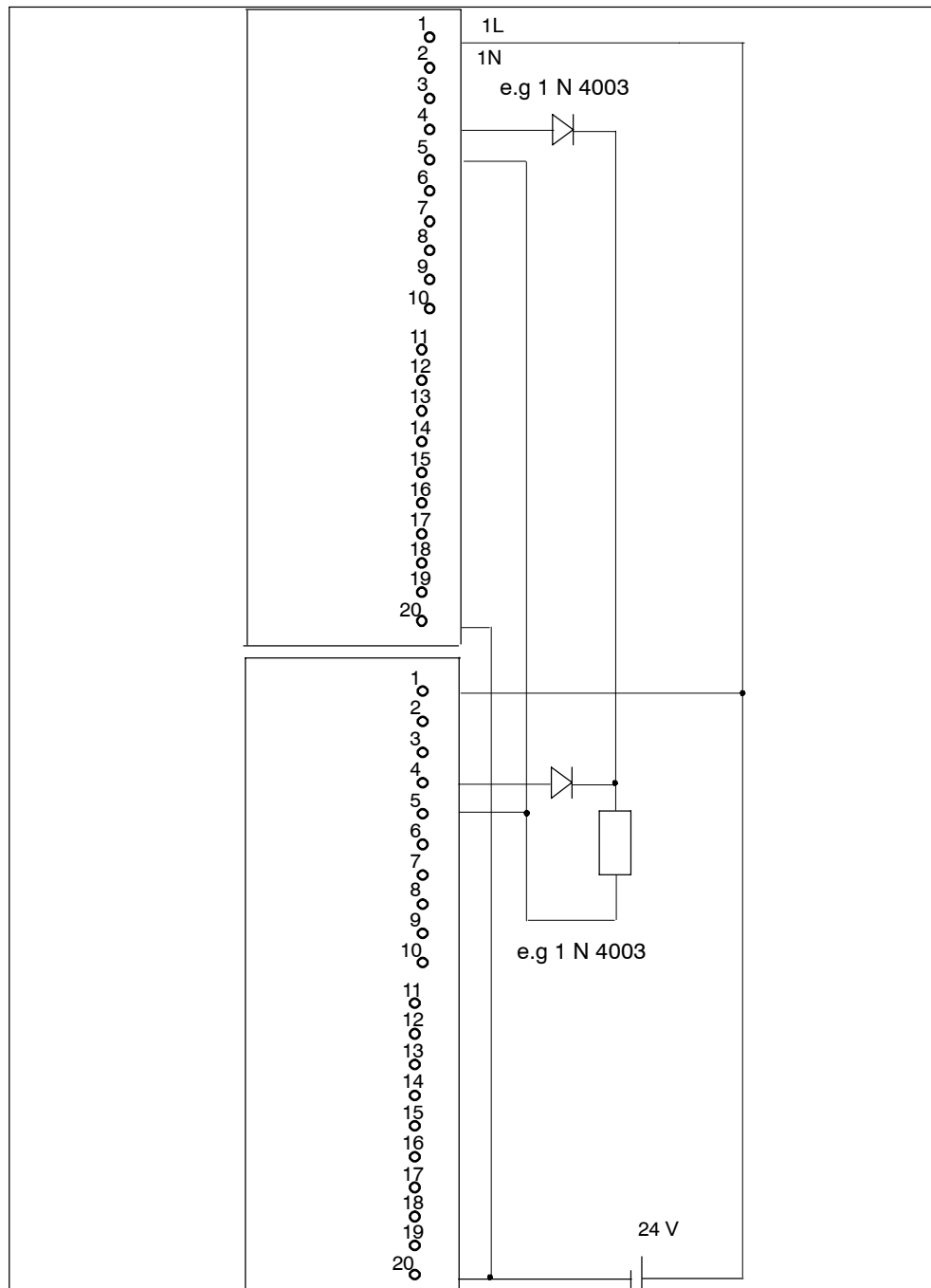
Figure F-17   Example of an interconnection with SM 322; DO 16 x DC 24 V/10 mA [EEx ib]

F.18   SM 322; DO 8 x DC 24 V/0.5 A, 6ES7 322-8BF00-0AB0

The diagram below shows the connection of an actuator to two redundant SM 322; DO 8 x DC 24 V/0.5 A. The actuator is connected to channel 0.
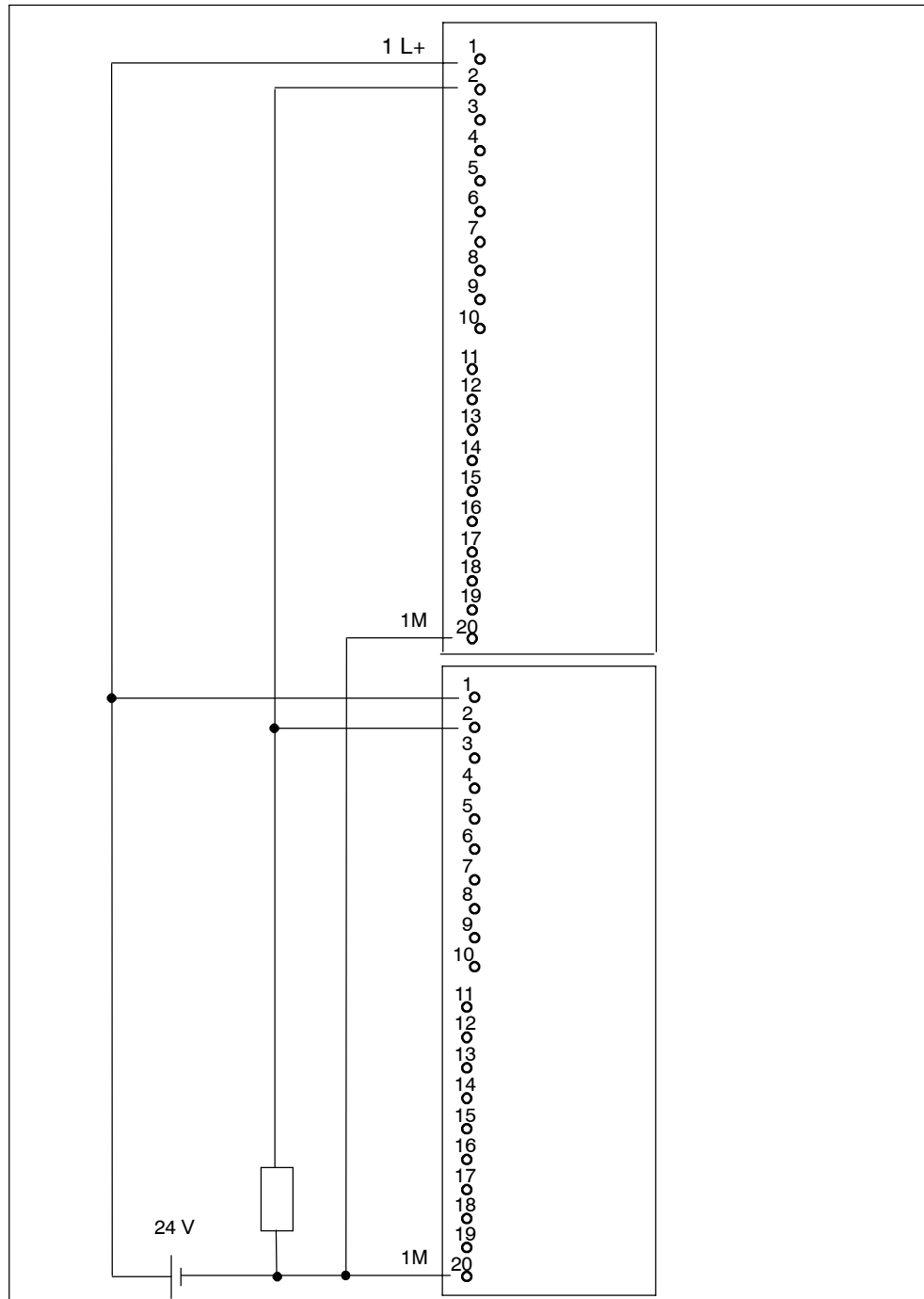
Figure F-18   Example of an interconnection with SM 322; DO 8 x DC 24 V/0.5 A

F.19 SM 322; DO 16 x DC 24 V/0.5 A, 6ES7 322-8BH01-0AB0

The diagram below shows the connection of an actuator to two redundant SM 322; DO 16 x DC 24 V/0.5 A. The actuator is connected to channel 8.
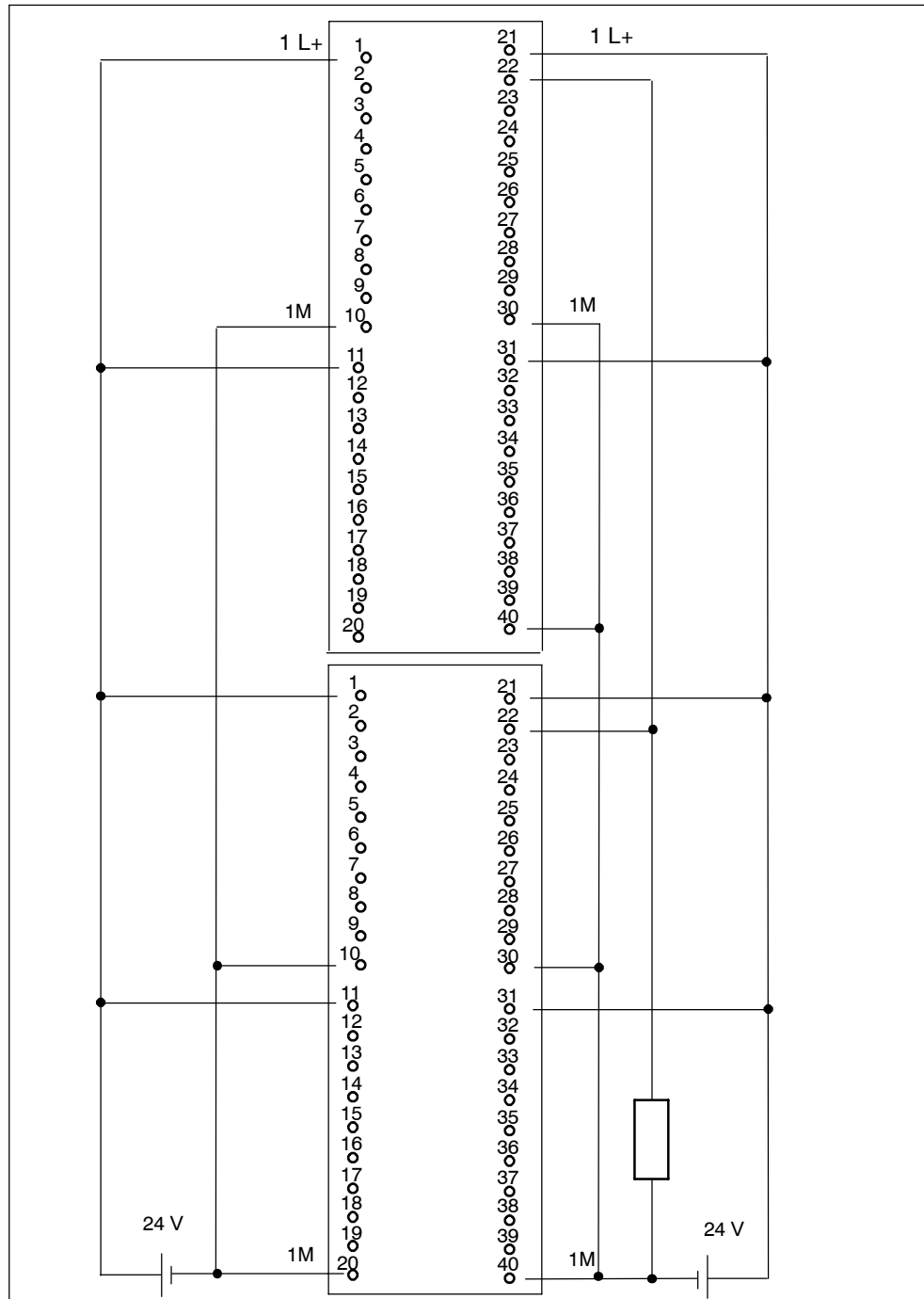
Figure F-19   Example of an interconnection with SM 322; DO 16 x DC 24 V/0.5 A

F.20 SM 332; AO 8 x 12 Bit; 6ES7 332-5HF00-0AB0

The diagram below shows the connection of two actuators to two redundant SM 332; AO 8 x 12 Bit. The actuators are connected to channels 0 and 4. Suitable diodes are those of the series 1N4003 ... 1N4007, or any other diode with $V_r >= 200$ V and $I_F >= 1$ A
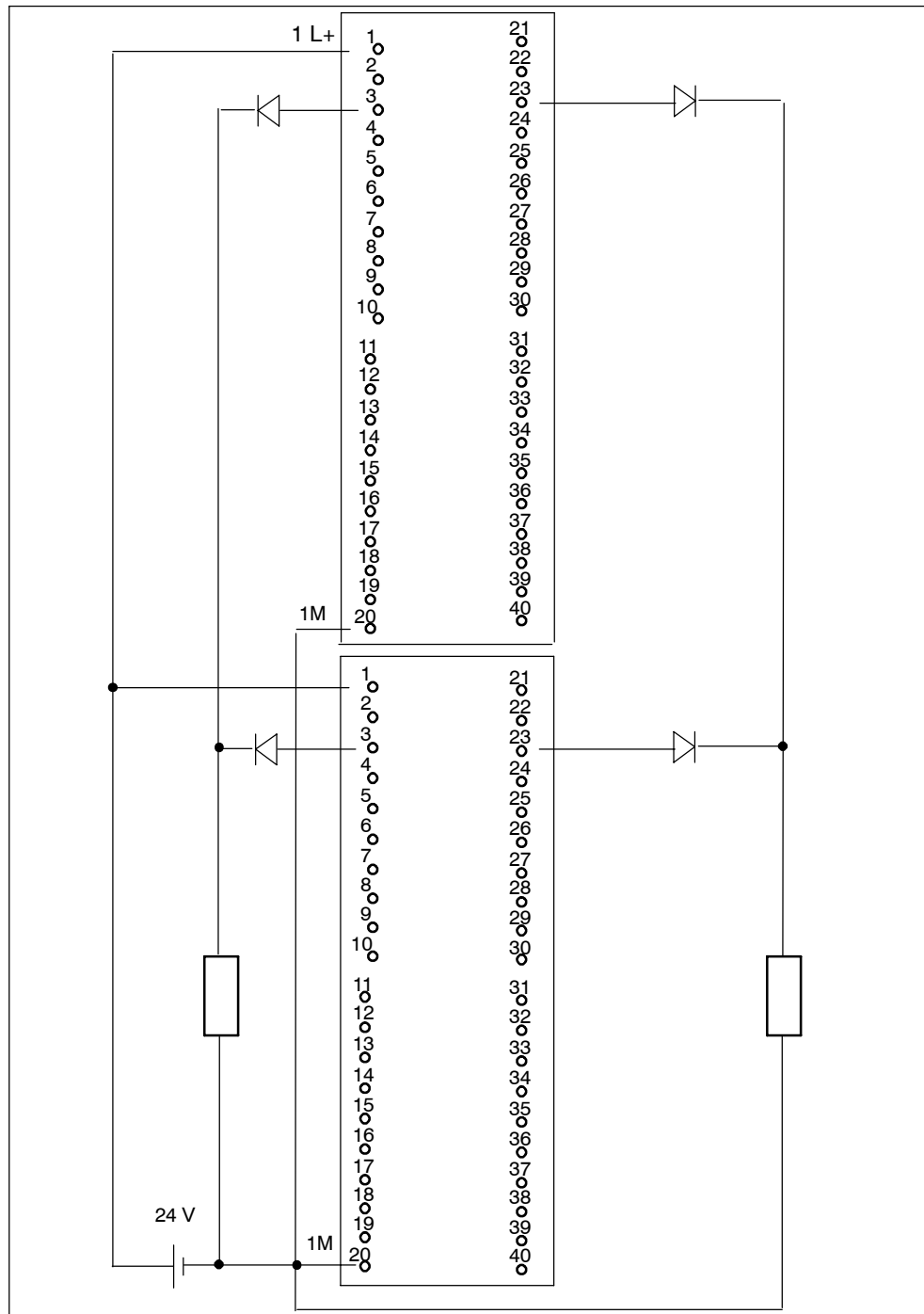
Figure F-20   Example of an interconnection with SM 332, AO 8 x 12 Bit

F.21 SM 332; AO 4 x 0/4...20 mA [EEx ib], 6ES7 332-5RD00-0AB0

The diagram below shows the connection of an actuator to two SM 332;
AO 4 x 0/4...20 mA [EEx ib]. The actuator is connected to channel 0.
Suitable diodes are those of the series 1N4003 ... 1N4007, or any other diode with
$V_r >= 200$ V and $I_F >= 1$ A
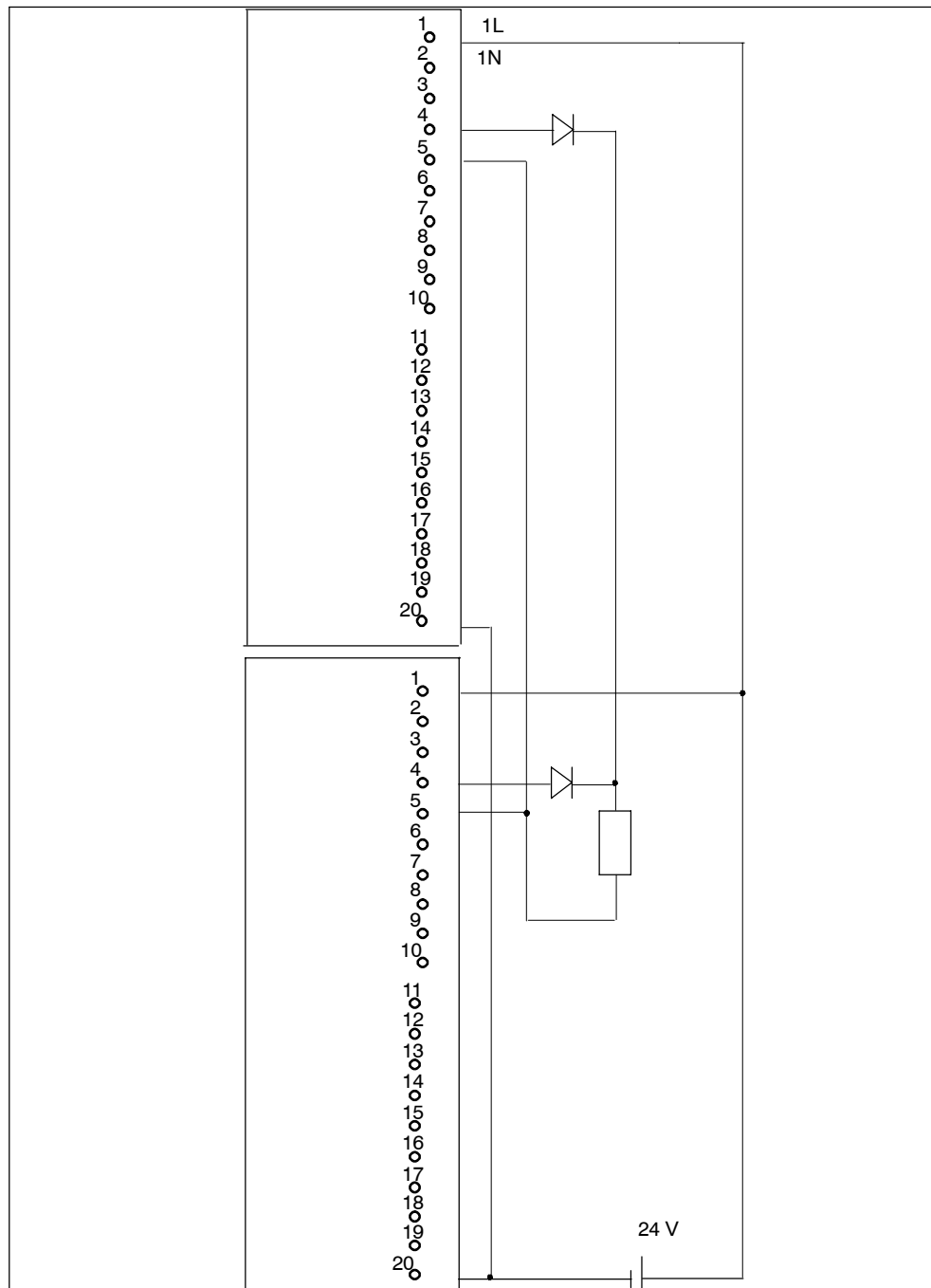
Figure F-21   Example of an interconnection with SM 332; AO 4 x 0/4...20 mA [EEx ib]

F.22 SM 422; DO 16 x AC 120/230 V/2 A, 6ES7 422-1FH00-0AA0

The diagram below shows the connection of an actuator to two
SM 422; DO 16 x 120/230 V/2 A. The actuator is connected to channel 0.
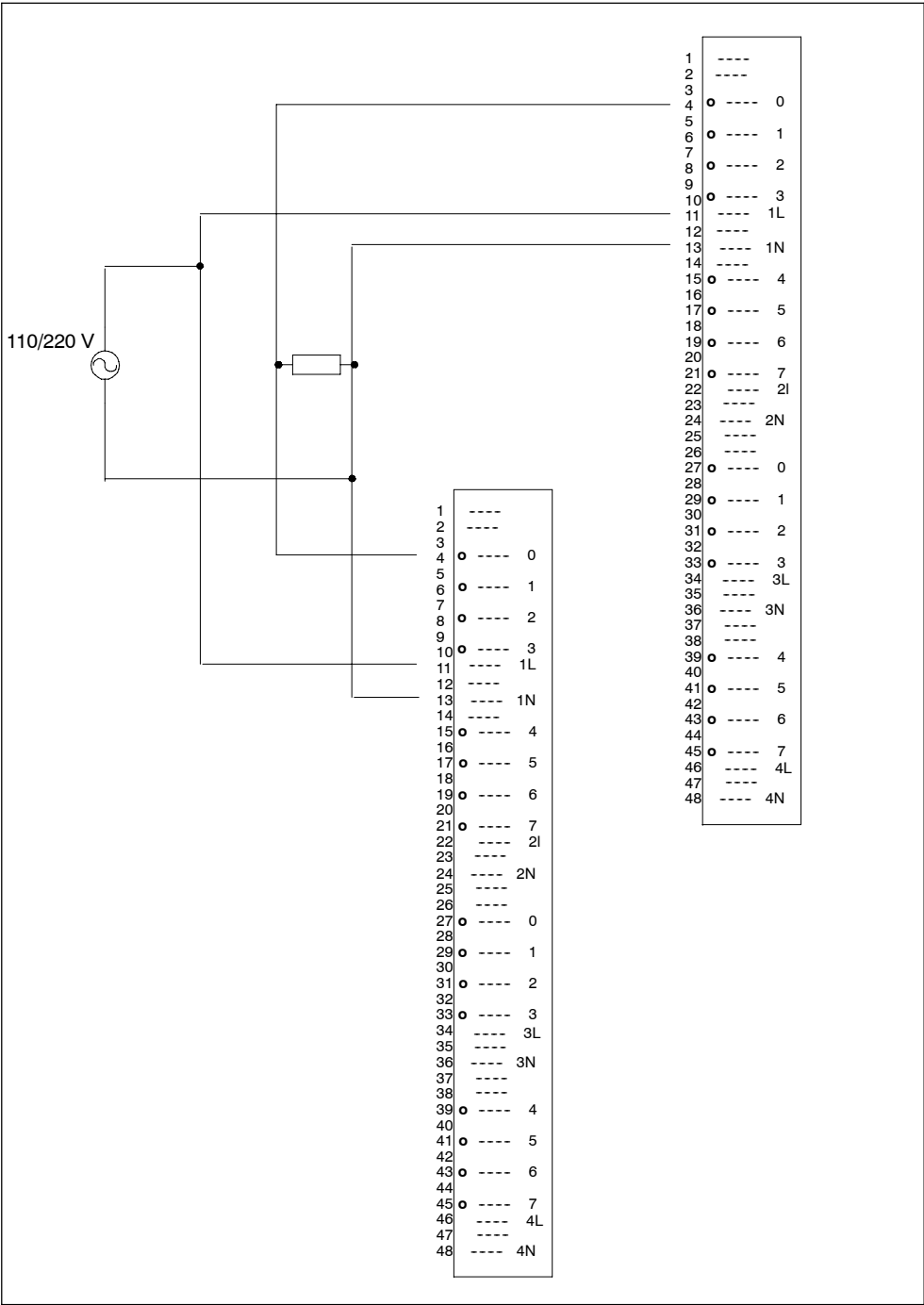
Figure F-22   Example of an interconnection with SM 422; DO 16 x 120/230 V/2 A

F.23 SM 422; DO 32 x DC 24 V/0.5 A, 6ES7 422-7BL00-0AB0

The diagram below shows the connection of an actuator to two SM 422; DO 32 x /24 V/0.5 A. The actuator is connected to channel 0. Suitable diodes are those of the series 1N4003 ... 1N4007, or any other diode with $V_r >= 200$ V and $I_F >= 1$ A

e.g 1 N 4003

e.g 1 N 4003

24 V
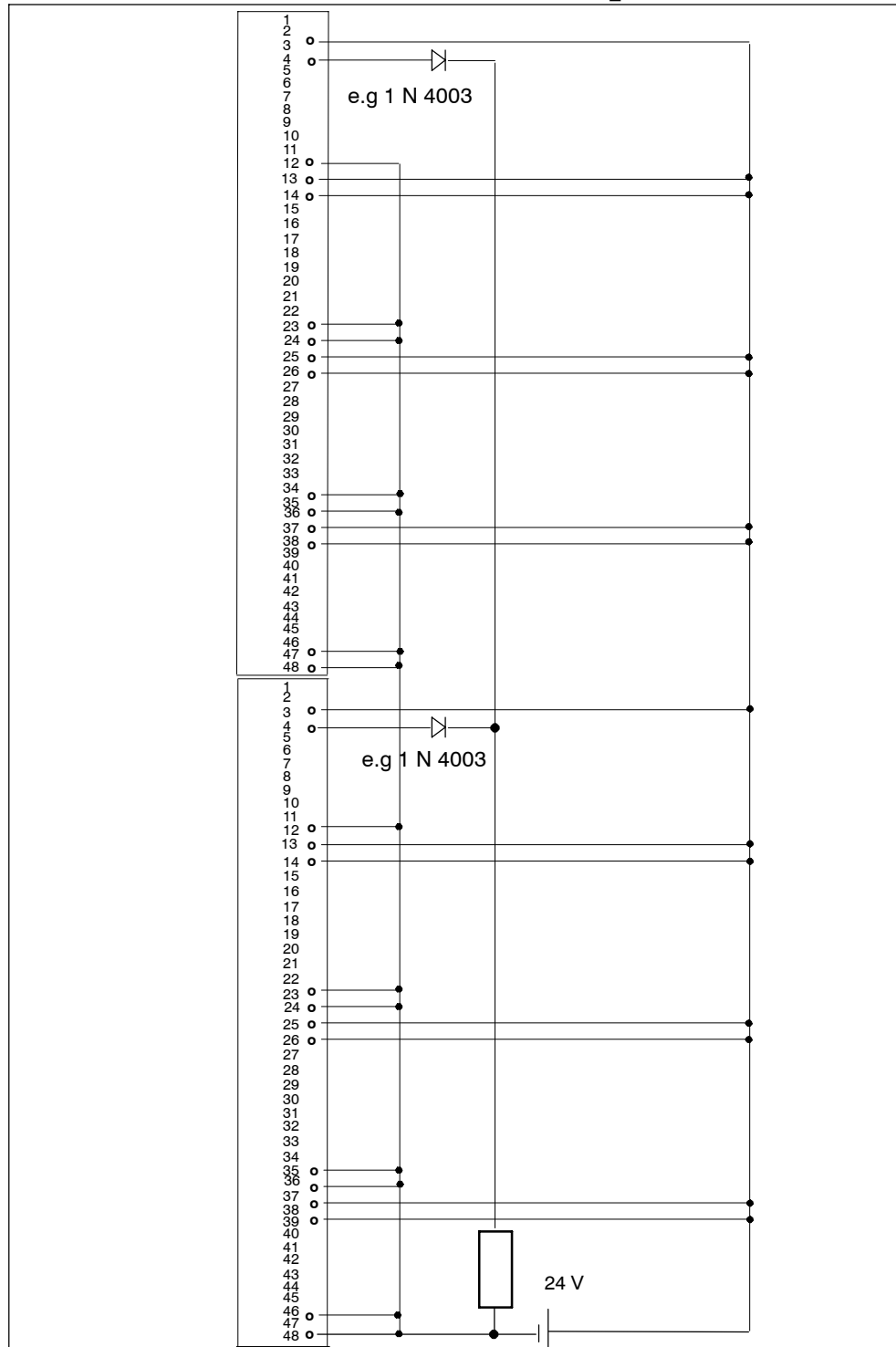
Figure F-23   Example of an interconnection with SM 422; DO 32 x DC 24 V/0.5 A

F.24   SM 331; AI 4 x 15 Bit [EEx ib]; 6ES7 331-7RD00-0AB0

The diagram below shows the connection of a 2-wire measuring transducer to two SM 331; AI 4 x 15 Bit [EEx ib]. The measuring transducer is connected to channel 1. Suitable Zener diode BZX85C6v2 or 1N4734A (6.2 V because of the 50 Ohm input resistance)
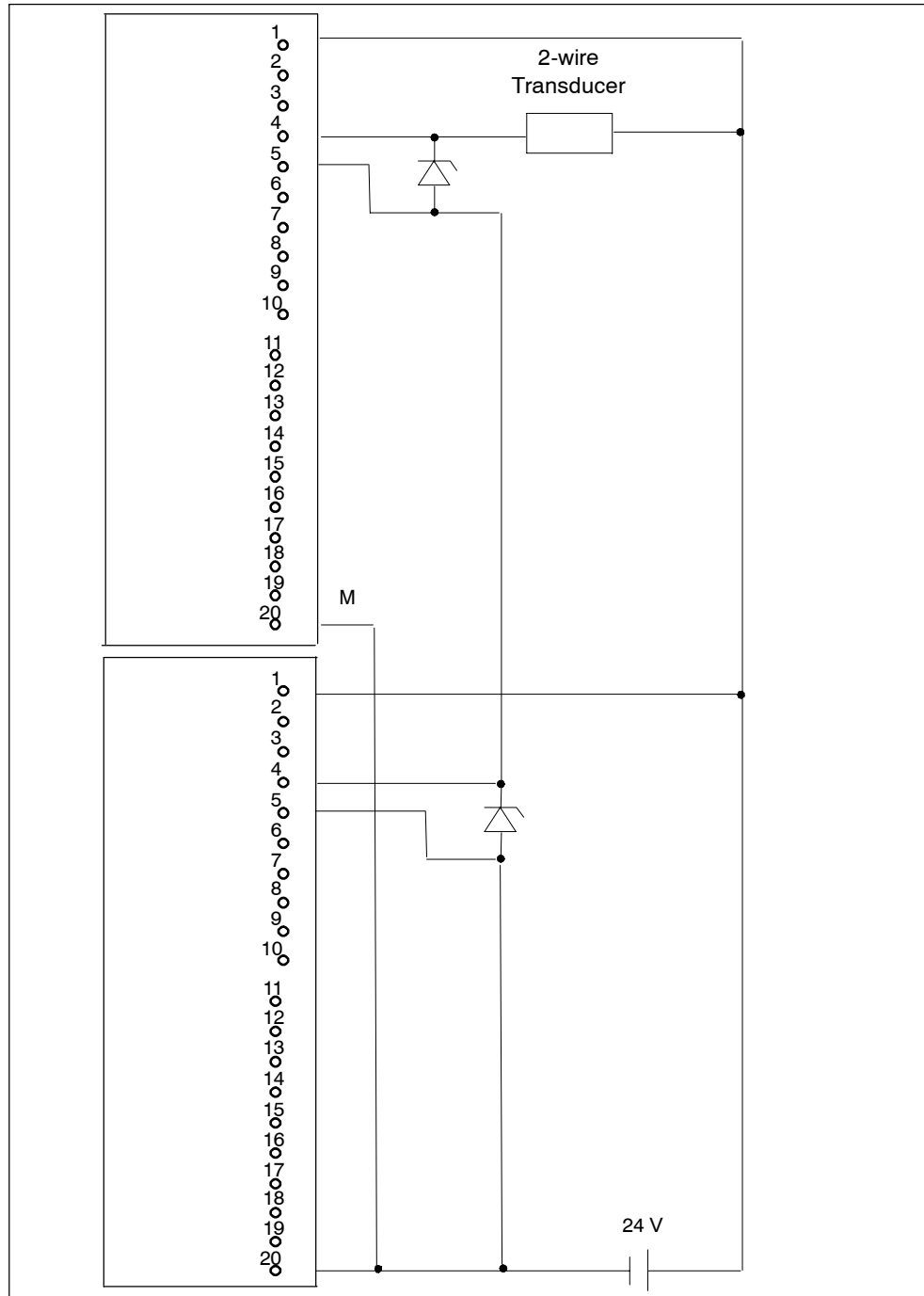
Figure F-24   Example of an interconnection with SM 331, AI 4 x 15 Bit [EEx ib]

F.25 SM 331; AI 8 x 12 Bit, 6ES7 331-7KF02-0AB0

The diagram below shows the connection of a measuring transducer to two SM 331; AI 8 x 12 Bit. The measuring transducer is connected to channel 1.
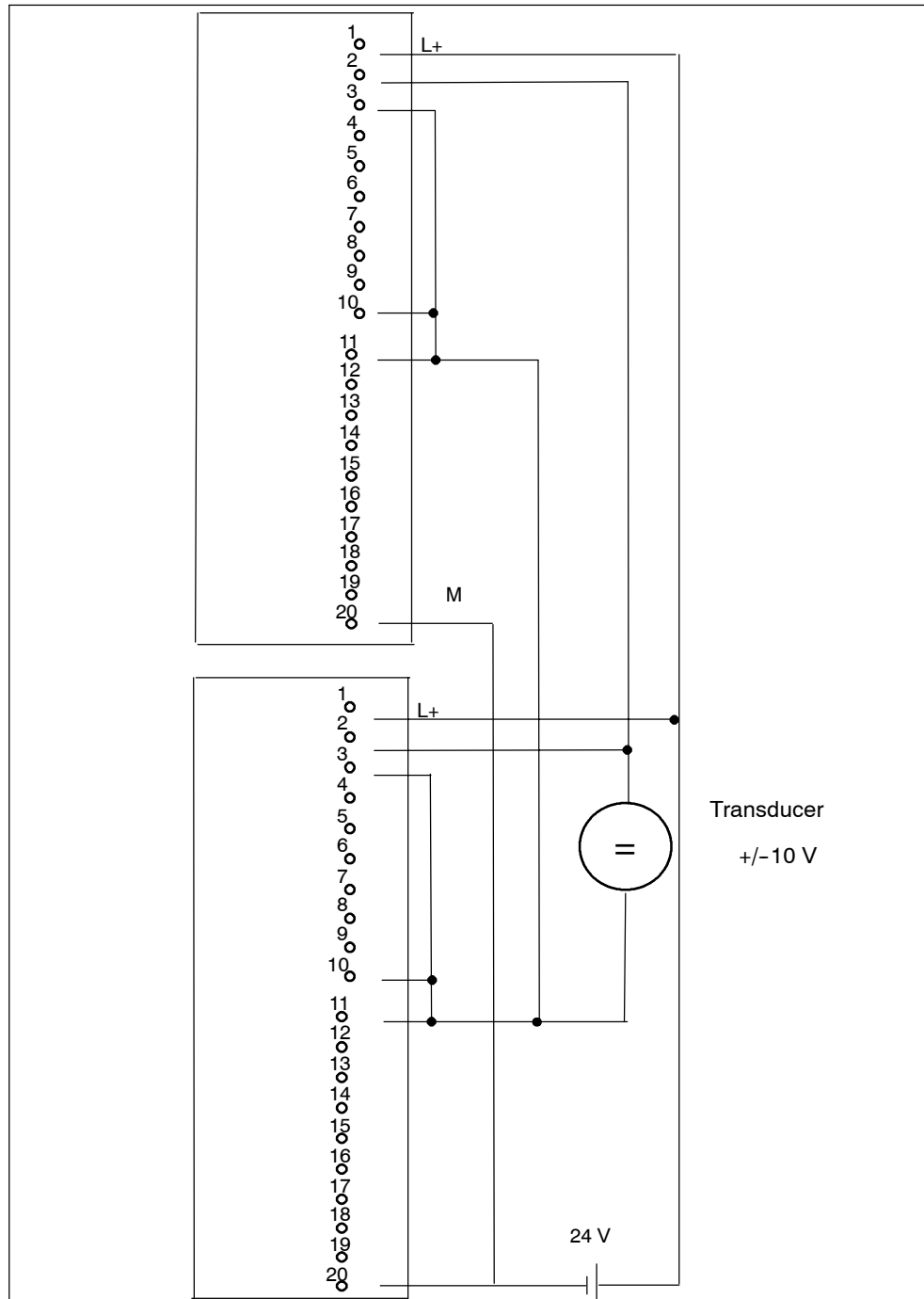
Figure F-25   Example of an interconnection with SM 331; AI 8 x 12 Bit

F.26   SM 331; AI 8 x 16 Bit, 6ES7 331-7NF00-0AB0

The diagram below shows the connection of a transmitter to two redundant
SM 331; AI 8 x 16 Bit. The transmitter is connected to channel3.
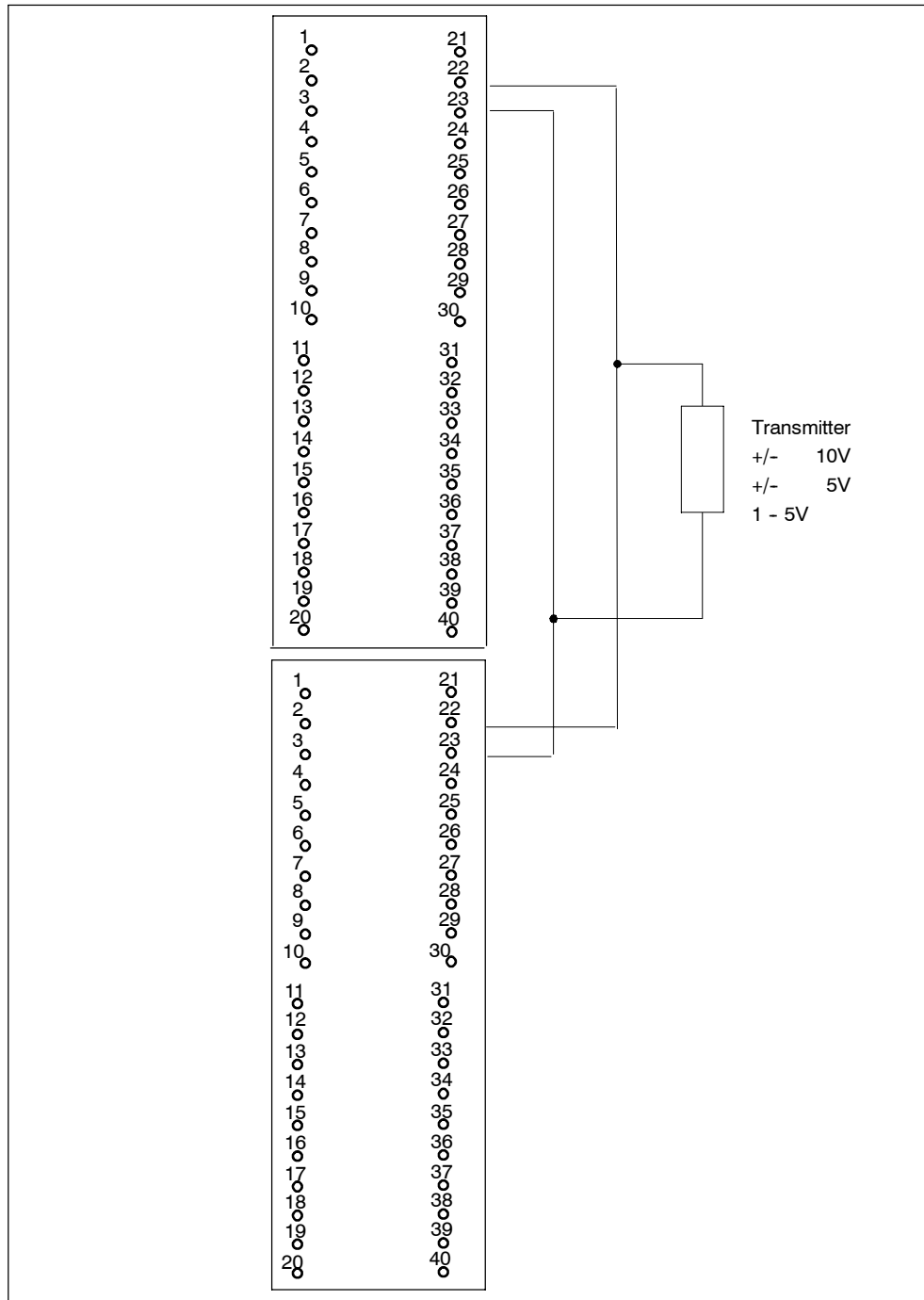
Figure F-26   Example of an interconnection with SM 331; AI 8 x 16 Bit

F.27 SM 332; AO 4 x 12 Bit; 6ES7 332-5HD01-0AB0

The diagram below shows the connection of an actuator to twoSM 332; AO 4 x 12 Bit. The actuator is connected to channel 0. Suitable diodes are those of the series 1N4003 ... 1N4007, or any other diode with $V_r >= 200$ V and $I_F >= 1$ A
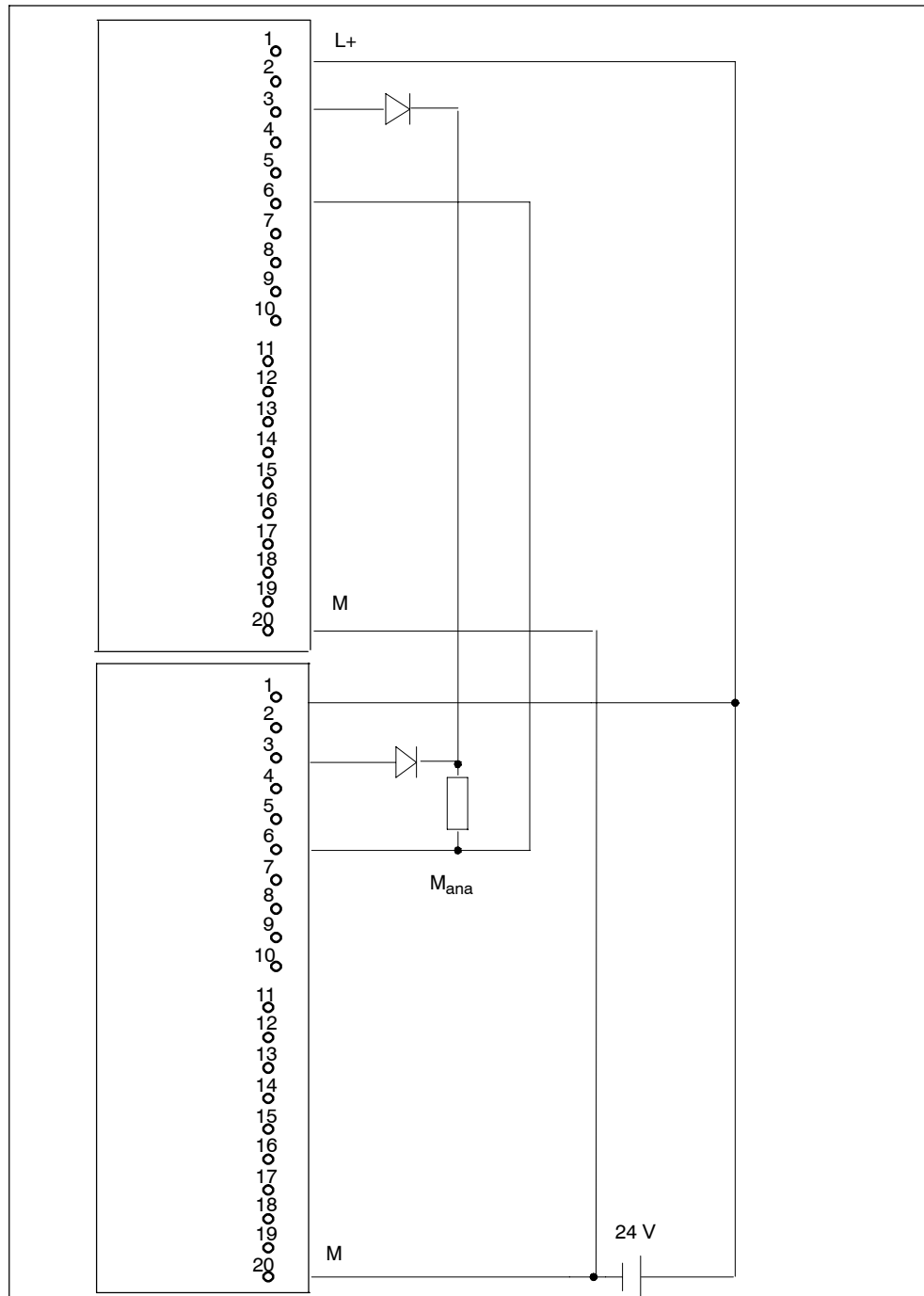
Figure F-27   Example of an interconnection with SM 332, AO 4 x 12 Bit

F.28 SM 431; AI 16 x 16 Bit, 6ES7 431-7QH00-0AB0

The diagram below shows the connection of a sensor to two SM 431;
AI 16 x 16 Bit. The sensor is connected to channel 0. Suitable Zener diode
BZX85C6v2 or 1N4734A (6.2 V because of the 50 Ohm input resistance)
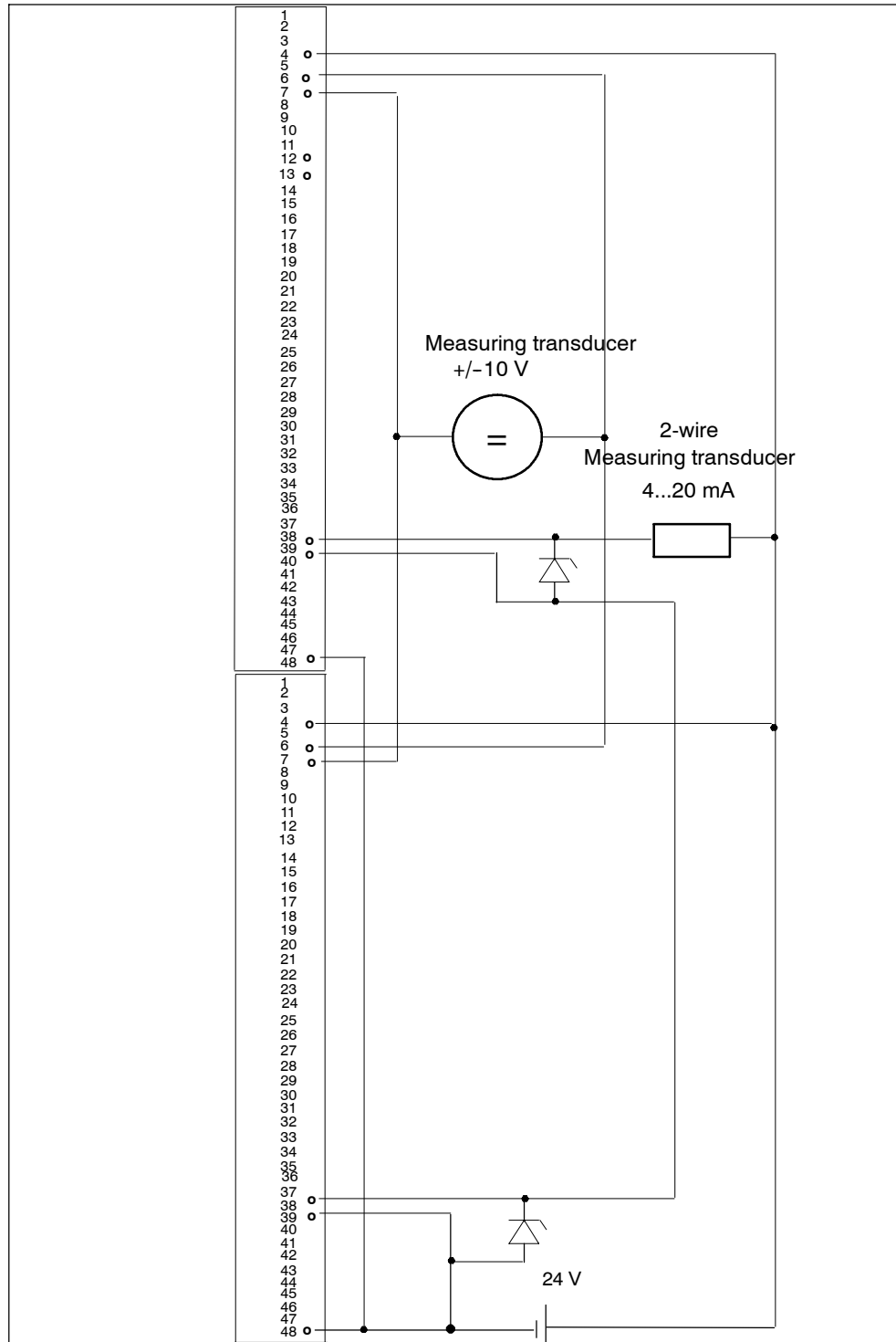
Figure F-28   Example of an interconnection with SM 431; AI 16 x 16 Bit

Glossary

1–out–of–2 system

    See **Dual–channel H system**

Comparison error

    An error that may occur while memories are being compared on a fault–tolerant system.

Dual–channel H system

    H system with two central modules

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 systems

    Fault–tolerant systems are designed to reduce production downtime. Availability can be enhanced, for example, by means of component redundancy.

H station

    A station containing two central processing units (master and standby).

H system

    Fault–tolerant system consisting of at least two central processing units (master and standby). The user program is processed identically in both the master and standby CPUs.

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 may be connected as one–sided or switched. Usage: "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 may be connected as one–sided or switched.

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).

Linking

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 loadmemory. 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 redundancy connection is lost. The user program is processed identically in both the master and standby CPUs.

Meantime between failures (MTBF)

The average time between two failures and, consequently, a criterion for the re-liability of a module or a system.

Meantime down time (MDT)

The mean down time **MDT** essentially consists of the time until error detection and the time required to repair or replace defective modules.

Meantime to repair (MTTR)

"Meantime to repair" denotes the average repair time of a module or a system, in other words, the time between the occurrence of an error and the time when 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 mode

In 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..

Redundant link

A link between the central processing units of a fault–tolerant system for synchronization and the exchange of data.

Self-test

In the case of fault–tolerant CPUs defined self–tests are executed during startup, cyclical processing and when comparison errors occur. They check the contents and the state of the CPU and the I/Os.

Single mode

In the single system mode of a fault–tolerant system, the master CPU is in RUN mode, whereas the standby CPU is in the STOP, TROUBLE-SHOOTING or DEFECTIVE mode.

Single operation

Referring to single operation, we mean the use of a fault-tolerant CPU in a standard SIMATIC-400 station.

Standby CPU

The redundant central processing unit of a fault–tolerant system that is linked to the master CPU. It goes to STOP mode when the redundancy connection is lost. The user program is processed identically 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 module

> An interface module to the redundant link on a fault–tolerant system

TROUBLESHOOTING

> An operating mode of the standby CPU of a fault–tolerant system in which the CPU performs a complete self-test..

Update

> In the update system mode of a fault–tolerant system, the master CPU updates the dynamic data of the standby CPU (synchronization).

Index

Numbers

41xH CPU
    DP address areas, 5-3
    DP master, 5-3

A

Address area, 41xH CPU, 5-3
Analog output signals, 8-33
Availability
    communications, 2-6
    definition, A-6
    I/O, 8-2
    of systems, 1-4

B

Base system, 2-3
bumpless continuation, 6-3
Bus connectors, 4-22
    MPI, 4-21
    PROFIBUS DP interface, 4-22
Bus interruption, 5-10
Bus topology, 5-5
BUSF, 5-6
BUSF1, 4-9
BUSF2, 4-9

C

calculation, Reaction time, 14-13
Central processing unit, 2-3
Change memory type, 12-46
Changeover to the CPU which contains the
    memory expansion, 7-14
Changeover to the CPU which contains the
    modified configuration, 7-13
Checksum error, 6-13
Cold restart, 4-15
    operating sequence, 4-16
Commissioning, 3-1
Commissioning the S7-400H, 3-4
Communication, 2-6
Communication blocks, consistency, 5-12
Communication functions, 7-12

Communication modules, E-1
Communication via MPI and C–bus, Cycle
    load, 14-4
Comparison error, 6-13
Components
    base system, 2-3
    duplicating, 1-4
Configuration , 2-1
Configuration types, I/O, 8-2
Configuring, 2-6, 10-2
Configuring networking, 10-7
Connection
    redundant S7, 9-3
    S7, 9-3
Consistent data access, 5-16
COUPLING, 6-9
Coupling, 7-2, 7-3, 7-4, 7-8, 7-16, 7-19, 8-42
    flow chart, 7-5
    monitoring periods, 8-42
    sequence, 7-8
    Special features, 7-29
    time–based reaction, 7-19
Coupling and update
    initiating, 7-4
    operation, 7-4
Coupling and update operations
    disabling, 7-16
    effects, 7-2
    performance, 7-28
Coupling with master/standby changeover, 7-8
Coupling, updating, 6-9
CPU
    mode selector switch, 4-12
    parameters, 4-23
CPU 414–4H, control and display elements,
    4-2
CPU 417–4H, control and display elements,
    4-2
CPU 41xH
    Siehe auch CPU 31x-2
    DP master, Diagnostics with LEDs, 5-6
CPU memory reset, 6-8
    sequence, 4-14
CPU–CPU communication, 4-21
Cycle control, execution time, 14-7
Cycle load, Communication via MPI and
    C–bus, 14-4