SIEMENS

SIMATIC

ET 200S
IM 151-7 CPU Interface Module

Manual

The following supplement is part of this documentation:

No.	Designation	Drawing number	Edition
1	Product information	A5E00385826-02	11/2005

This manual has the order number:
6ES7151-1AB00-8BA0

Edition 11/2003
A5E00058783-04

Safety Guidelines

This manual contains notices intended to ensure personal safety, as well as to protect the products and connected equipment against damage. These notices are highlighted by the symbols shown below and graded according to severity by the following texts:

Danger

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

Warning

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

Caution

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

Caution

indicates that property damage can result if proper precautions are not taken.

Notice

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

Qualified Personnel

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

Correct Usage

Note the following:

Warning

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

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

Trademarks

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

Third parties using for their own purposes any other names in this document which refer to trademarks might infringe upon the rights of the trademark owners.

Siemens AG
Bereich Automation and Drives
Geschaeftsgebiet Industrial Automation Systems
Postfach 4848, D- 90327 Nuernberg

Preface

Purpose of the manual

This manual supplements the *ET 200S Distributed I/O System* manual. It describes all the functions of the IM 151-7 CPU interface module. The manual does not deal with general ET 200S functions. You will find these in the *ET 200X Distributed I/O System* manual (see also the Section *"Integration in the information landscape"*).

The information contained in this manual and in the *ET 200S Distributed I/O System* manual will enable you to operate the ET 200S with the IM 151-7 CPU interface module as an I slave on the PROFIBUS-DP or in an MPI network. The master functionality in combination with the DP master module is also described.

Required level of knowledge

Knowledge of the field of automation engineering is required to understand the manual.

Knowledge on how to use computers or other PC equipment (e.g. programming devices) under the Windows 95/98/2000 and NT operating system is also required. You should also be familiar with the *STEP 7* basic software. Refer to the "Programming with STEP 7 V5.x" manual.

Scope of validity of the manual

This manual applies to the interface module IM 151-7 CPU with the order number 6ES7 151-7AA10-0AB0 and to the DP master module with the order number 6ES7 138-4HA00-0AB0, as well as to the components of the ET 200S distributed I/O system specified in the *ET 200S Distributed I/O System* manual.

This manual contains a description of the components that were valid at the time the manual was published. We reserve the right to enclose a Product Information bulletin containing up-to-date information about new components and new versions of components.

For IM 151-7 CPU with the order number 6ES7 151-**7AA00**-0AB0 and IM 151-7 CPU FO with the order number 6ES7 151-**7AB00**-0AB0, please download the corresponding manual, *ET 200S Interface Module IM 151/CPU* from the Internet:
`http://www.siemens.com/automation/service&support`
ID 2460607.
This manual is also availabe in the SIMATIC Manual Collection.

Changes compared to the previous version

The following changes/additions have been made since the previous version of the manual:

- The order number or, respectively, the documentation package's packet assembly has been changed (see also the section "Integration in the information landscape").
- Minor changes have been made.

Standards, certificates and approvals

The ET 200S distributed I/O system is based on the IEC 61784-1:2002 Ed1 CP 3/1 standard.
The ET 200S distributed I/O system fulfills the requirements and criteria of IEC 61131, Part 2 and the requirements for obtaining the CE marking. The ET 200S has certificates and approvals for CSA, UL, FM, and shipbuilding.

You will find detailed information on these standards, certificates and approvals in the *ET 200S Distributed I/O System* manual.

Position in the information landscape

The following list shows a summary of the documentation packages or manuals:

ET 200S Distributed I/O System

6ES7151-1AA10-8xA0[1]

- Installing and wiring the ET 200S
- Commissioning and diagnostics for the ET 200S
- Technical specifications of the IM151-1, digital and analog electronic modules
- Order numbers for the ET 200S

ET 200S Motor Starters

- Installing and wiring motor starters
- Commissioning and diagnostics for motor starters
- Technical specifications of motor starters
- Order numbers for motor starters

ET 200S interface module IM151-7 CPU

6ES7151-1AB00-8xA0[1]

- Addressing of the IM151-7 CPU
- ET 200S with IM151-7 CPU in the PROFIBUS network
- Commissioning and diagnostics for the IM151-7 CPU
- Technical specifications of the IM151-7 CPU

***Automation system S7-300,* list of operations**

6ES7398-8AA10-8xN0[1]

- ..
- IM 151-7 CPU
- ...

ET 200S Process-Related Functions
6ES7151-1AC00-8xA0[1]

- 1Count 24V/100kHz
- 1Count 5V/500kHz
- 1SSI
- 2PULSE

ET 200S Positioning

6ES7151-1AD00-8xA0[1]

- 1STEP 5V/204kHz
- 1POS INC/Digital
- 1POS SSI/Digital
- 1POS INC/Analog
- 1POS SSI/Analog

Serial interface module ET 200S

6ES7151-1AE00-8xA0[1]

- 1SI 3964/ASCII
- 1SI MODBUS/USS

[1] x= language designation for order numbers

The documentation packages or manuals can only be ordered in the languages German and English. In addition, the languages French, Spanish and Italian are available in the Internet (see Service & Support in the Internet)

Note

The *ET 200S fail-safe modules* manual is part of the *S7 F Systems* documentation package.

Guide to the manual

You can quickly access specific information in the manual by using the following aids:

- At the start of the manual you will find a complete table of contents and a list of the diagrams and tables that appear in the manual.
- An overview of the contents of each section is provided in the left-hand column on each page of each chapter.
- Important technical terminology used in the manual is defined in the Glossary.
- At the end of the manual you will find a comprehensive index enabling rapid access to the information you are looking for.
- Language designation for the order numbers of the manuals, e.g. 6ES7151-1AA00-8xA0

  x = A = German, B = English

Special note

In addition to the ET 200S manuals, you will also need the manual for the DP master used and the documentation for the configuration and programming software used (see the list in Appendix A of the *ET 200S Distributed I/O System* manual).

Note

You will find a detailed list of the contents of the ET 200S manuals in Section 1.2 of this manual.

We recommend that you begin by reading this section so as to find out which parts of which manuals are most relevant to you in helping you to do what you want to do.

Recycling and disposal

The components of the IM 151-7 CPU contain very few harmful substances which means that the unit can be recycled.
To ensure environment-friendly recycling and disposal of old equipment, contact an officially approved disposal company that deals with electronics scrap.

Additional support

Please contact your local Siemens representative if you have any queries about the products described in this manual.

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

Training center

We offer training courses to help familiarize you with the ET 200S distributed I/O system and the SIMATIC S7 programmable controller. Please contact your local training center or the central training center in

D 90327 Nürnberg.
Telephone: +49 (911) 895-3200
Internet: `http://www.sitrain.com`

A&D Technical Support

Accessible throughout the world at any time of day:

Technical Support

Worldwide (Nuremberg) **Technical Support** Loc. time: 0:00 to 24:00 / 365 days Teleph.: +49 (0) 180 5050-222 Fax: +49 (0) 180 5050-223 E-mail: adsupport@ siemens.com GMT: +1:00		
Europe/Africa (Nuremberg) **Authorization** Loc. time: Mon. - Fri. 8:00 to 17:00 Teleph.: +49 (0) 180 5050-222 Fax: +49 (0) 180 5050-223 E-mail: adsupport@ siemens.com GMT: +1:00	**United States** (Johnson City) **Technical Support and Authorization** Loc. time: Mon. - Fri. 8:00 to 17:00 Teleph.: +1 (0) 423 262 2522 Fax: +1 (0) 423 262 2289 E-mail: simatic.hotline@ sea.siemens.com GMT: -5:00	**Asia / Australia** (Beijing) **Technical Support and Authorization** Loc. time: Mon. - Fri. 8:00 to 17:00 Teleph.: +86 10 64 75 75 75 Fax: +86 10 64 74 74 74 E-mail: adsupport.asia@ siemens.com GMT: +8:00
Technical support and authorization staff generally speak English and German.		

Service & Support on the Internet

In addition to our documentation, we also offer you all of our know-how online on the Internet.

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

There you will find:

- the newsletter, which constantly supplies you with the latest information on your products
- a search function in Service & Support to help you find the documents you need
- a forum in which users and specialists can exchange their experiences worldwide
- your local contact partner for Automation & Drives in our contact database
- information on on-site service, repairs and spare parts. You will find a lot more information under ”Services”.

Contents

Figures

Tables

Product Overview

1

In this chapter

The product overview provides information about

- The role of the IM 151-7 CPU interface module within the ET 200S distributed I/O system.
- Which manuals in the ET 200S manual package contain what information.

Chapter overview

1.1 What is the IM 151-7 CPU interface module?

What is the IM 151-7 CPU?

The IM 151-7 CPU is a component of the ET 200S distributed I/O system with degree of protection IP 20. The IM 151-7 CPU interface module is an ”intelligent pre-processing unit” (I slave). It enables you to decentralize control tasks.

An ET 200S with an IM 151-7 CPU can therefore exercise full and, if necessary, independent control over a process-related functional unit and can be used as a stand-alone CPU. The IM 151-7 CPU also features DP master functionality in combination with the DP master module. The use of the IM 151-7 CPU leads to further modularization and standardization of process-related functional units and simple, clear machine concepts.

How is the IM 151-7 CPU integrated in the ET 200S?

The IM 151-7 CPU interface module is integrated in the ET 200S in the same way as any other module. In other words, its configuration concept, installation and expansion capability are the same.

Limitations in the use of ET 200S modules

The following modules can be used with the IM 151-7 CPU as of the version specified here:

Table 1-1 Limitations in the use of ET 200S modules

Module	Order number	As of product version	Product release
1COUNT 24V/100kHz	6ES7 138-4DA03-0AB0	01	07/2002
1COUNT 5V/500kHz	6ES7 138-4DE01-0AB0	01	07/2002
1SSI	6ES7 138-4DB01-0AB0	01	07/2002
EM 1STEP 5V/204kHz	6ES7 138-4DC00-0AB0	04	04/2001
1 SI 3964/ASCII	6ES7 138-4DF00-0AB0	02	08/2000
1SI MODBUS/USS	6ES7 138-4DF10-0AB0	02	06/2002
2AI U HF	6ES7 134-4LB00-0AB0	03	03/2001
2AI I 2/4WIRE HF	6ES7 134-4MB00-0AB0	02	02/2000
2AO U HF	6ES7 135-4LB01-0AB0	01	11/2002

View

The figure below shows a sample configuration of an ET 200S with an IM 151-7 CPU.

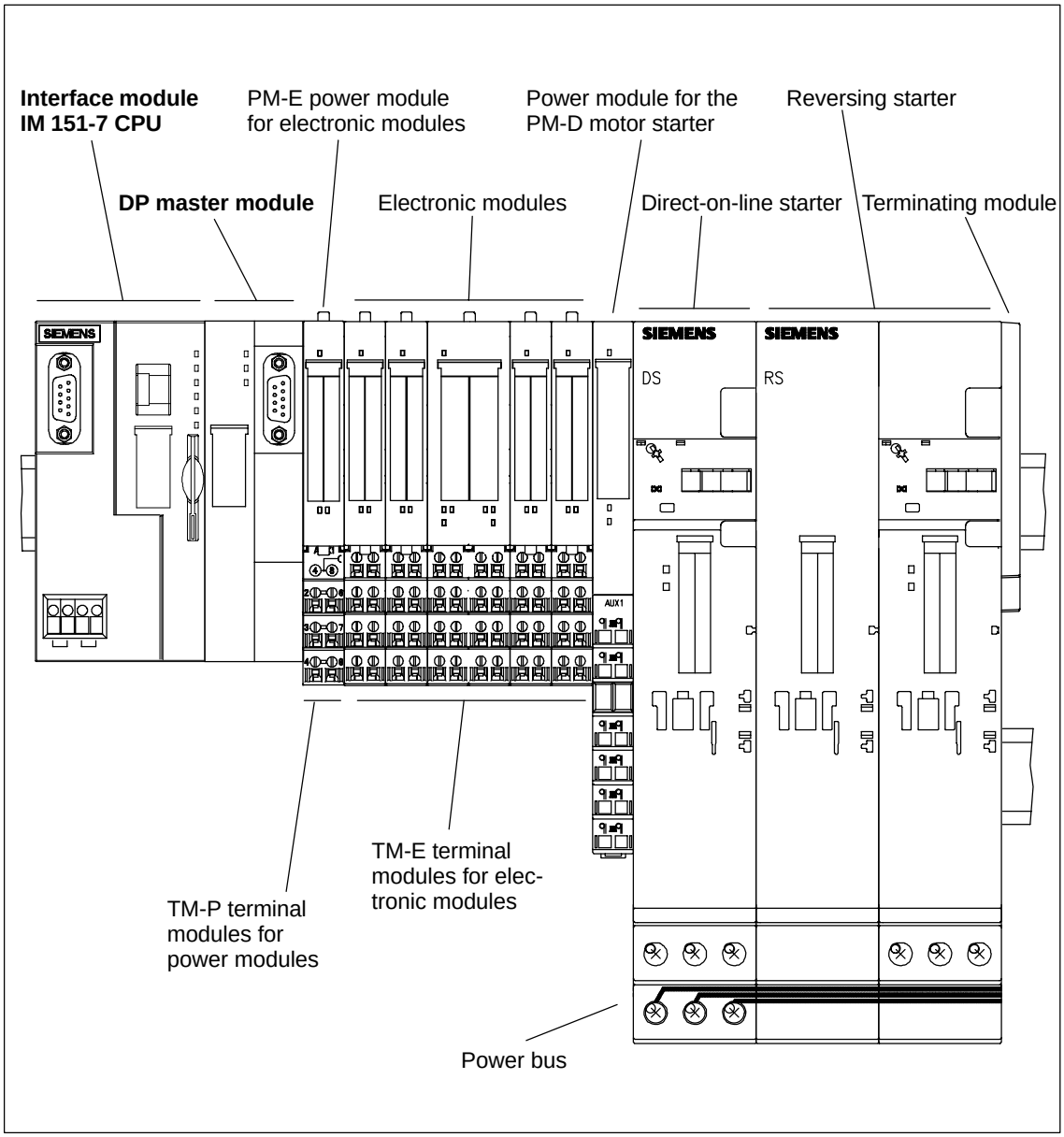


Figure 1-1 View of the ET 200S distributed I/O system with the IM 151-7 CPU and the DP master module

Features of the IM 151-7 CPU compared to other modules

The IM 151-7 CPU interface module has the following special features:

- The interface has PLC functionality (integrated CPU component with 48 kByte working memory).
- The interface module can only be operated with fitted load memory (MMC).
- The interface module can be enhanced with up to 63 I/O modules from the ET 200S range.
- The interface module has a mode selector with positions for RUN, STOP and MRES.
- There are 6 LEDs on the front of the interface module to indicate the following:
  - ET 200S faults (SF)
  - Bus faults (BF)
  - Supply voltage for electronic components (ON)
  - Force requests (FRCE)
  - Operating mode of the IM 151-7 CPU (RUN and STOP)
- Connection to the PROFIBUS-DP via RS 485
- In combination with the DP master module, the IM 151-7 CPU can be used as a DP master.

How is the ET 200S configured with the IM 151-7 CPU?

To configure the ET 200S with IM 151-7 CPU (configuration and parameterization), you will need the configuration software *STEP 7*

- as of V5.1 + Service Pack 4 for IM 151-7 CPU as an I slave
- as of V5.2 + Service Pack 1 for DP master functionality (IM 151-7 CPU with DP master module)

How to configure the ET 200S with IM 151-7 CPU is described in Chapters 7.1 and 6.2 of this manual.

How is the IM 151-7 CPU programmed?

To program the IM 151-7 CPU, you will need the configuration software *STEP 7*

- as of V5.1 + Service Pack 4 for IM 151-7 CPU as an I slave
- as of V5.2 + Service Pack 1 for DP master functionality (IM 151-7 CPU with DP master module)

The *Instruction list* contains the *STEP 7* instruction set for programming the IM 151-7 CPU.

1.2 Guide to the ET 200S manuals

They utilize the following components ...

The components of ET 200S are described in various manuals. They are parts of various documentation packages. The figure below shows possible configuration variants of the ET 200S and the necessary manuals in the documentation packages.

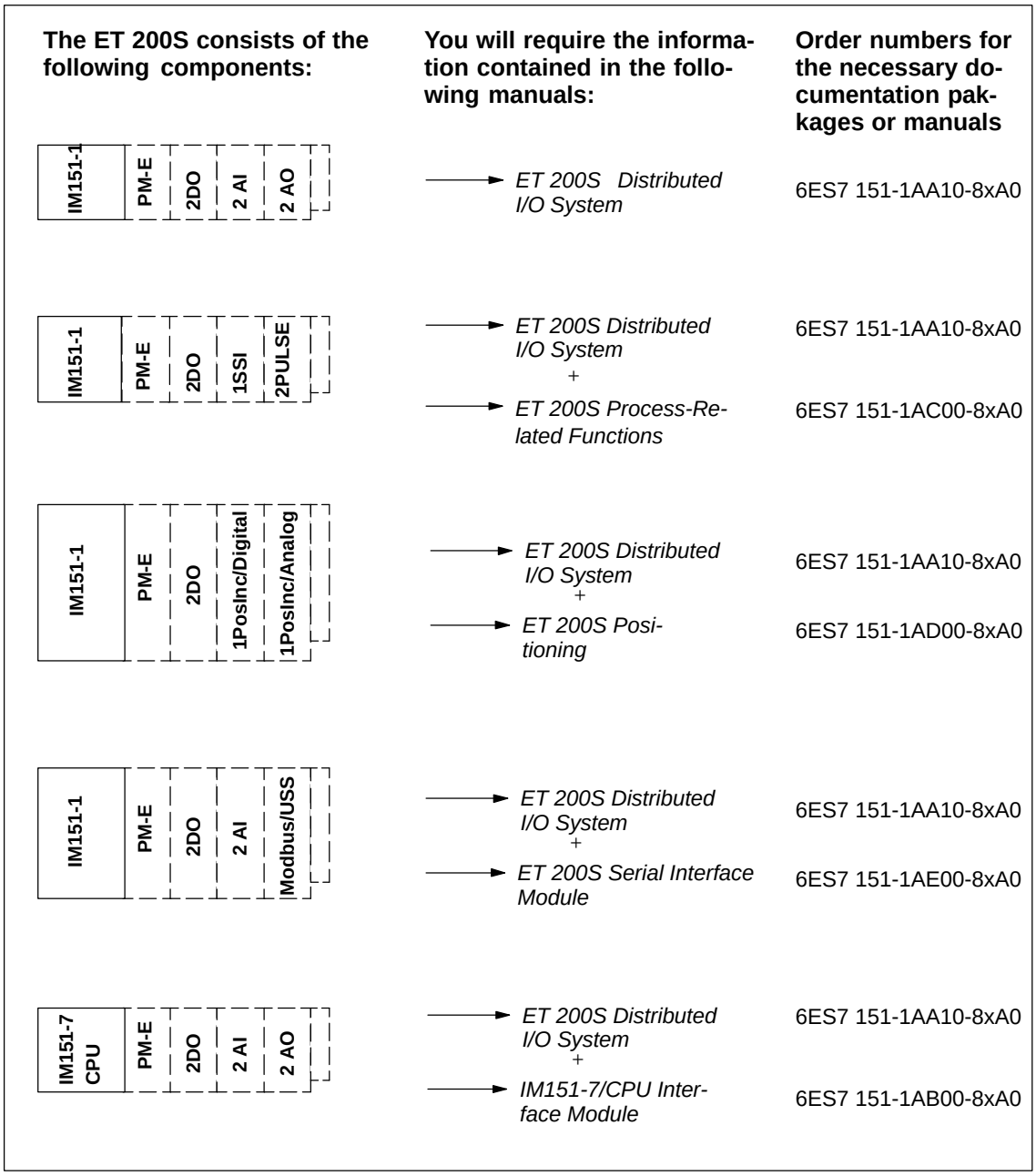


Figure 1-2 Components and the manuals required for them

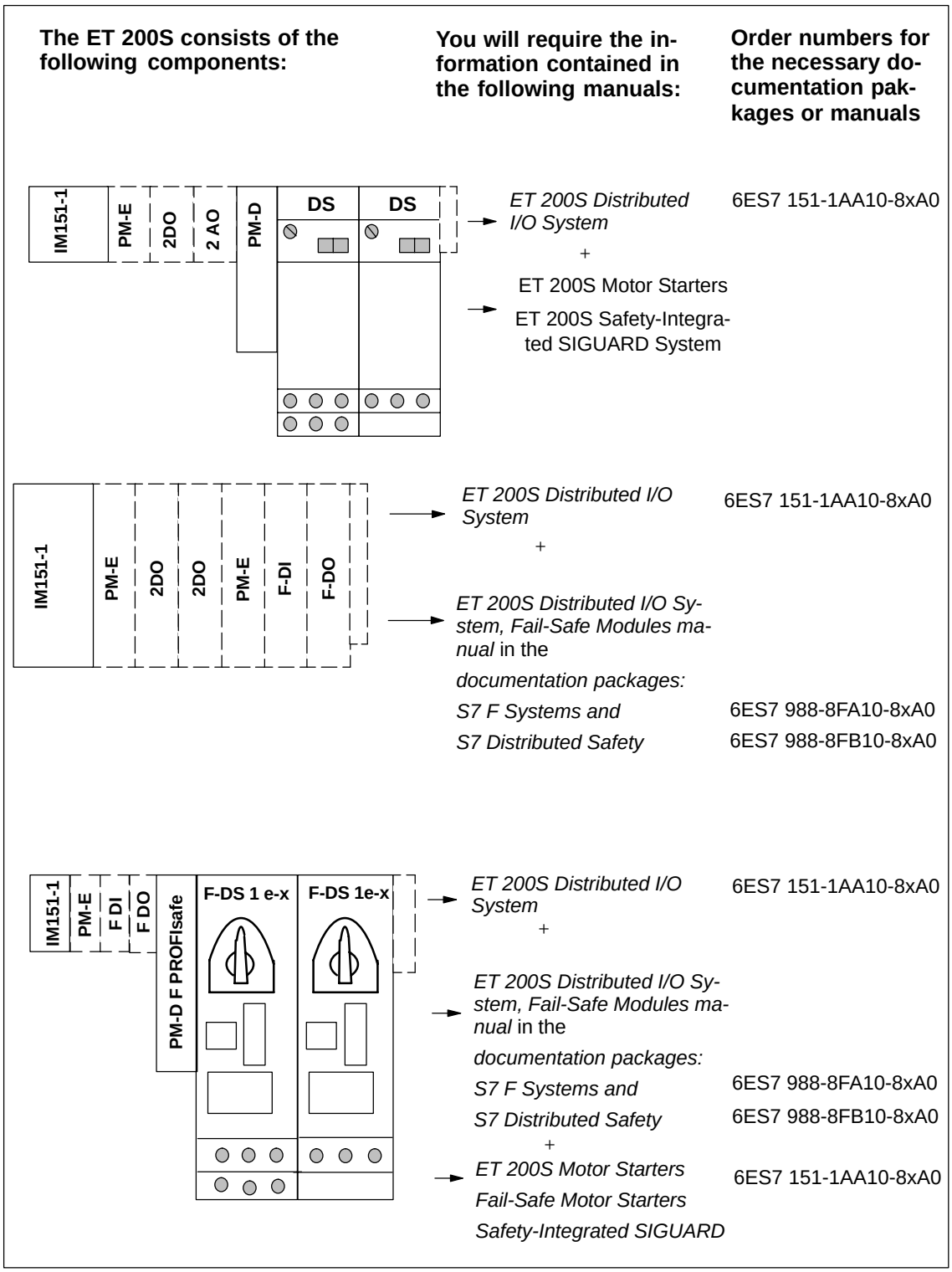


Figure 1-3 Components and the manuals required for them (continued)

Where do you find what information?

The following table will help you find the information you require quickly. It tells you which manual you need to refer to and which section deals with the topic you are interested in.

Table 1-2 Topics of the manuals in the ET 200S manual package

Description	Manual								
	Distributed I/O System ET 200S	ET 200S Motor Starter Fail-Safe Motor Starter	Interface Module IM 151-7 CPU	Technological Functions ET 200S	Positioning ET 2200S	Serial interface Module ET 200S	Distributed I/O System ET 200S fail-safe modules	Safety-integrated System description	S7 Distributed Safety Configuring and programming
ET 200S components	1	1	1				2	2	2
Configuration possibilities	1	3	4				3	3	1
Communication								4	
Configuration	3							7	3
Addressing			2				5		
Installation	2	4					5		
Electrical configuration and wiring of the ET 200S		5					6		
Programming								8	5
Commissioning and diagnostics	3	6					7		
Functions			5						
General technical specifications	4	7					8		
Technical specifications			6	2-5	2	2,3			
Terminal modules	5, 9, 11	9							
Power modules	6, 9-11	10							
Direct starters and soft starters	7								
Reversible starters	8								
Safety-integrated ET 200S SIGUARD	9								
Interface module		8							
Electronic modules		11-15							

Table 1-2 Topics of the manuals in the ET 200S manual package

Description	Manual								
	Distributed I/O System ET 200S	ET 200S Motor Starter Fail-Safe Motor Starter	Interface Module IM 151-7 CPU	Technological Functions ET 200S	Positioning ET 2200S	Serial interface Module ET 200S	Distributed I/O System ET 200S fail-safe modules	Safety-integrated System description	S7 Distributed Safety Configuring and programming
Positioning module					3-6				
Expansion module	10								
Fail-safe modules	11						9		
Monitoring, cycle and reaction times			7				12	9	
Order numbers	A	A					11		
Dimension drawings	B	B					10		
Applications	C								
Glossary	GI	GI	GI				13	10	9

The frame for configuration and parameter assignment for the IM 151-7 CPU can be found on the Internet at http://www.ad.siemens.de/simatic-cs

ET 200S fail-safe modules

You can find the *ET 200S Distributed I/O System, Fail-Safe Modules* manual in the *S7 F Systems* (order number 6ES7 988-8FA10-8xA0) documentation package and in the *S7 Distributed Safety* documentation package (order number 6ES7 988-8FB10-8xA0).

Getting Started

2

Introduction

This guide takes you through the 10 commissioning steps required to set up a functioning IM 151-7 CPU application by running through a concrete example. In this way, you will get to know the basic functions of your IM 151-7 CPU for the following:

- Hardware and software
- Stand-alone operation (MPI)
- Intelligent DP slave (PROFIBUS-DP)

Prerequisites

You must be familiar with the fundamentals of electronic/electrical engineering and have experience of working with computers and Microsoft® Windows™ 95/98/NT/2000.

Danger

The IM 151-7 CPU, the ET 200S and the S7-300 are used in installations and systems that require you to comply with specific rules and regulations that vary depending on the application.
Please note the relevant safety and accident prevention regulations, such as IEC 204 (emergency stop systems).
Non-compliance with these regulations can result in serious injury and damage to both machinery and equipment.

Chapter overview

Required material and tools

Quantity	Article	Order number (SIEMENS)
1	S7-300 system, consisting of power supply (PS), CPU with DP interface (here: CPU 315 2-DP), digital input module (DI) in slot 4 and digital output module (DO) in slot 5, incl. mounting rail, bus connectors and cabling	various
1	Power supply (PS), e.g. PS 307 with power connection cable (optional)	e.g. 6ES7 307-1EA00-0AA0
1	IM 151-7 CPU with terminating module	e.g. 6ES7 151-7AA10-0AB0
1	SIMATIC Micro Memory Card (MMC)	e.g. 6ES7 953-8LL00-0AA0
1	Power module (PM)	e.g. 6ES7 138-4CA00-0AA0
1	Digital input module (DI)	e.g. 6ES7 131-4BD00-0AA0
1	Digital output module (DO)	e.g. 6ES7 132-4BD00-0AA0
1	Terminal module (TM) for the PM	e.g. 6ES7 193-4CC30-0AA0
2	Terminal modules for DI and DO	e.g. 6ES7 193-4CB30-0AA0
1	Mounting rail for the ET 200S	various
1	Programming device (PD) with PROFIBUS-DP interface, installed software STEP 7 Version ≥ 5.1 and PD cable (up to 1.5 MBit/s)	various
1	PROFIBUS-DP cable	various
1	Screwdriver with tip width 3 mm	commercially available
1	Screwdriver with tip width 4.5 mm	commercially available
1	Diagonal cutting pliers and tools for wire stripping	commercially available
1	Tool for pressing on wire end ferrules	commercially available
approx. 2 m	Stranded wire with 1 mm^2 cross section with appropriate wire end ferrules, type A, length 6 mm and 12 mm	commercially available
4	1-pin ON button (24 V)	commercially available

2.1 1st step: Installing the IM 151-7 CPU (ET 200S) and S7-300

Se-quence	Description
1	Install the S7-300 as described in *Installation Manual for Automation System S7-300, Setting Up*.
2	If you want to operate the IM 151-7 CPU using a dedicated power supply, hang the PS in the mounting rail of the S7-300 and push in until it engages.
3	Hang the IM 151-7 CPU in the mounting rail and push in until it engages.
4	Hang the TM for the PM in the mounting rail to the right of the IM 151-7 CPU and push in until it engages.
5	Push the TM to the left until it engages audibly in the IM 151-7 CPU.
6	Repeat points 3 and 4 for two TMs for the electronics modules and finally for the terminating module (does not engage in the mounting rail).
7	Push the PM into the appropriate TM until it engages.
8	Push the DI into the free TM on the left until it engages.
9	Push the DO into the last free TM until it engages.
10	Insert the micro memory card into the IM 151-7 CPU (must be installed otherwise the system will not work). A micro memory card with unknown content should be erased beforehand at the programming device.

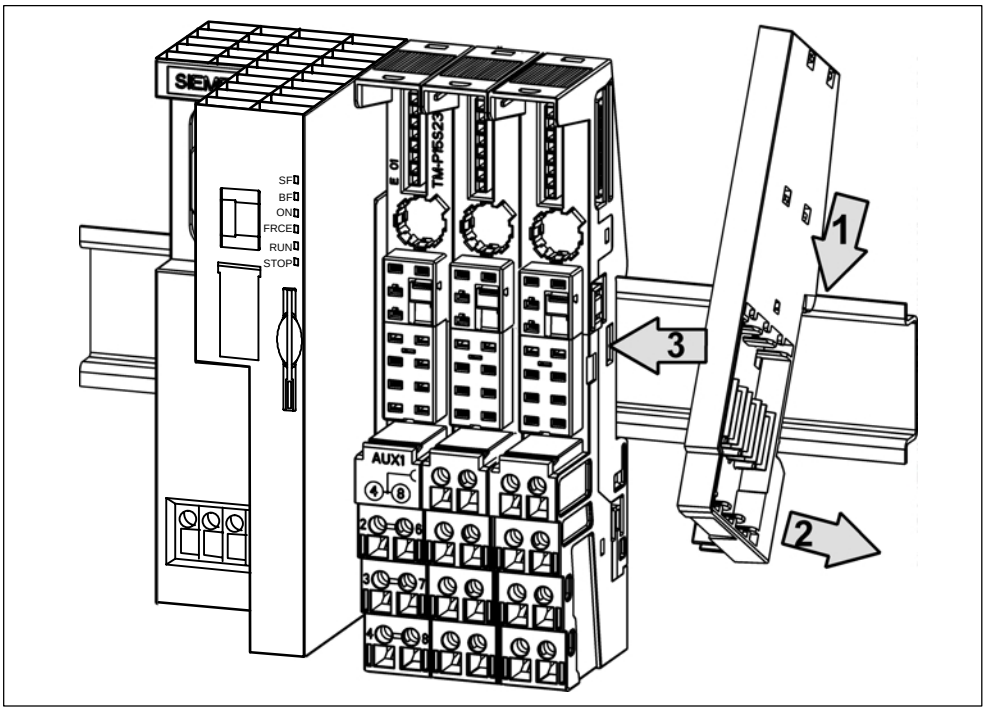


Figure 2-1 Installing the IM 151-7 CPU (ET 200S)

2.2 2nd step: Wiring the IM 151-7 CPU (ET 200S) and S7-300

Se-quence	Description
1	Wire the S7-300 as described in *Installation Manual for Automation System S7-300, Setting Up*.
2	Lengthen the connections for each of the 4 buttons using a cable. Strip 6 mm of insulation from the free cable ends and cap the ends with wire end ferrules.
3	At the DI of the S7-300, connect each of the inputs 1.1 (terminal 13) and 1.2 (terminal 14) to L+ on the PS of the S7-300 using a button.
4	Connect the two remaining 1-pin buttons to the DI of the ET 200S as follows: • connect one button to terminals 1 and 3 • connect the other button to terminals 5 and 7 **Note regarding spring terminals** Releasing the spring of a connection: Insert a screwdriver with 3 mm tip as far as it will go into the upper round hole of the terminal, pulling the screwdriver handle upwards slightly if necessary. A free cable end can then be inserted into the square hole below. Pull the screwdriver back out again and check that the cable is fitted securely.
5	Wire terminal 2 on the TM of the PM to L+ of the PS and terminal 3 on the TM of the PM to M of the PS. The ends of the cables to be connected must be stripped by 11 mm and capped with wire end ferrules.
6	Wire terminal 1L+ of the IM 151-7 CPU to L+ of the PS and terminal 1M of the IM 151-7 CPU to M of the PS. **Note** • The ends of the cables to be connected must be stripped by 11 mm and capped with wire end ferrules. • The PS of the S7-300 can also be used to supply power to the IM 151-7 CPU and to the PM.
7	Connect the PD and IM 151-7 CPU to the PD cable and tighten all connectors.
8	Connect the PS of the ET 200S, the PS of the S7-300 and the PD to the mains power supply system.

Illustration showing the S7-300 (the power supply wiring for the DI and DO is not shown; the PD is connected to the S7-300)

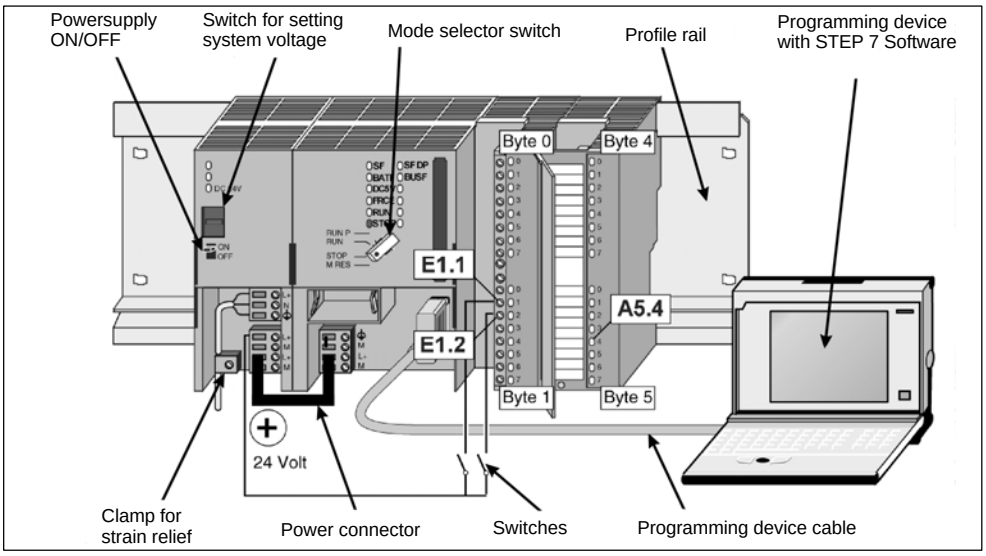
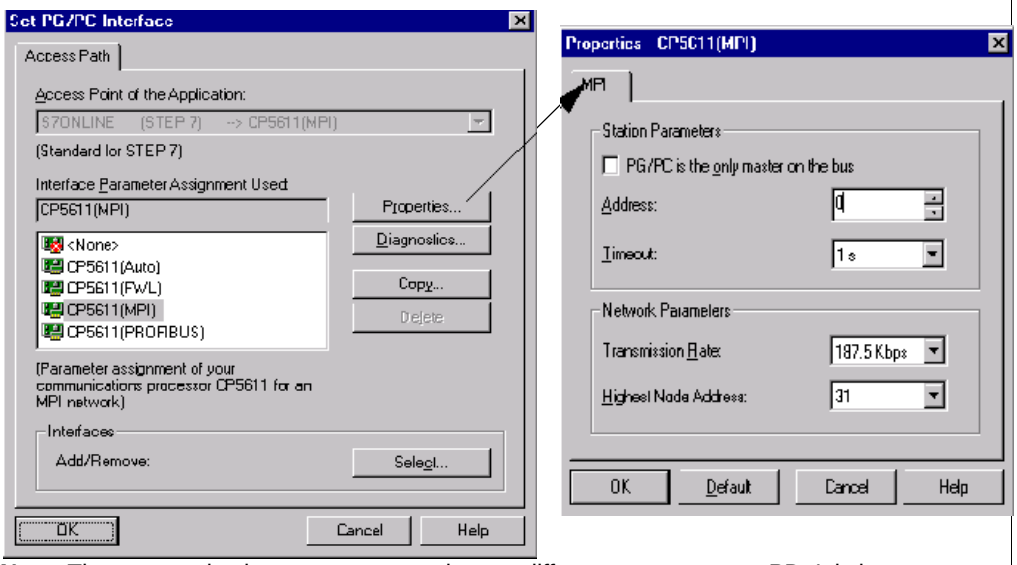


Figure 2-2 Illustration showing the S7-300

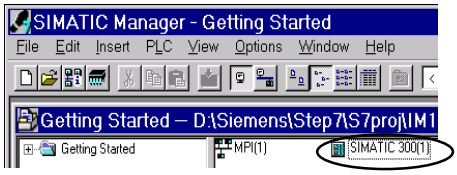
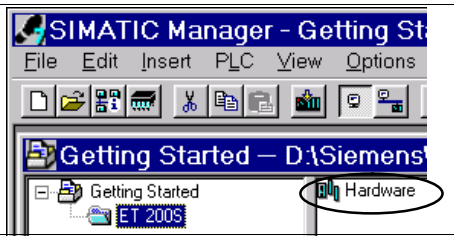
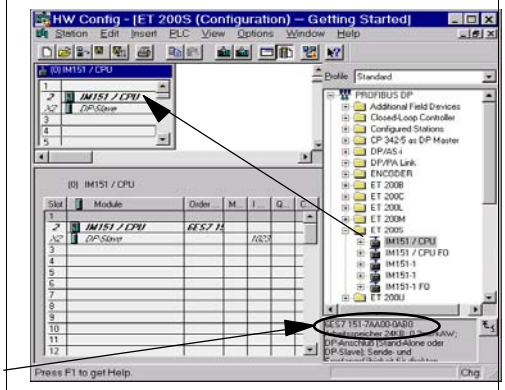
2.3 3rd step: Commissioning the IM 151-7 CPU (ET 200S)

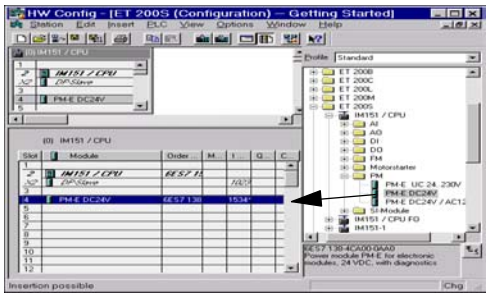
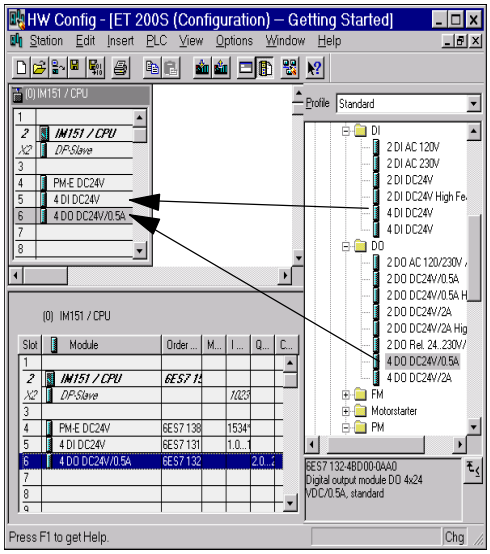
Note

On **initial commissioning** (condition when supplied) of the ET 200S, the CPU can be accessed using the MPI address 2, HSA 31 and 187.5 kBaud.

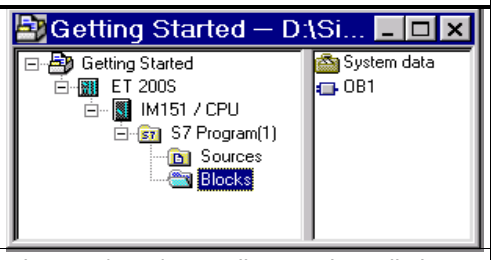
Se-quence	Description
1	Switch on the PS of the IM 151-7 CPU. **Result:** • The *24 VDC* LED lights up on the PS. • The *PWR* and *SF* LEDs light up on the PM. • All LEDs light up on the IM 151-7 CPU, the *SF*, *BF*, *FRCE* and *RUN* LEDs go out again and the *STOP* LED starts to flash rapidly. The IM 151-7 CPU performs a memory reset.
2	Then press the two buttons which are connected to the DI module. The *1* LED lights up when the buttons at terminal 1 and 3 are pressed. The *5* LED lights up when the buttons at terminal 5 and 7 are pressed.
3	Switch on your PD and start the SIMATIC Manager on the Windows Desktop.
4a	In the main menu of the SIMATIC Manager, click on Tools and select the menu item **Set PD/PC Interface**. Configure the PD/PC interface as follows:
4b	 **Note:** The communication processor may have a different name on your PD. It is important to make sure that the MPI version is set.
5	Confirm the settings with OK and close the ”Set PD/PC Interface” program.

2.4 4ht step: Configuring the IM 151-7 CPU for stand-alone operation (MPI)

Se-quence	Action	Result
1	Does the wizard for a new project appear in the SIMATIC Manager?	If yes: Close the wizard because the IM 151-7 CPU is not supported by the project wizard. If no: Proceed to point 2
2	In the main menu of the SIMATIC Manager, navigate to **File** and select the menu item **New.** Enter ”Getting Started” as the project name and click on the **OK** button.	A new project is created and opened.
3	Navigate to **Insert** and select the menu item **Station.** In the list, click on: **SIMATIC 300 Station**.	
4	Rename this station to ”ET 200S”	”SIMATIC 300(1)” is renamed to ”ET 200S”.
5	In the SIMATIC Manager, navigate to ET 200S Station. Double-click on the **Hardware** symbol in the right-hand part of the window to open the Editor used to edit the hardware configuration.	
6	If no catalog is shown in the right-hand part of the window, activate it by selecting the command **Catalog** in the **View** menu. In the catalog, navigate through **PROFIBUS-DP** to **ET 200S.** Drag the **IM 151-7 CPU** whose order number matches the order number on your IM 151-7 CPU, and drop it into the top left-hand window. By default, the IM 151-7 CPU is integrated as a stand-alone CPU (MPI/not networked). **Note:** You can find out the order number in the catalog by clicking on an **IM 151-7 CPU** in the catalog. The order number of this IM 151-7 CPU then appears in the field below the catalog.	

Se-quence	Action	Result
7	Navigate through the appropriate **IM 151-7 CPU** to **PM**. Drag the **PM** whose order number matches the order number on your PM, and drop it onto slot 4.	
8	Repeat point 8 for the DI (slot 5) and DO (slot 6)	
9	Select the command **Save and compile in the Station** menu.	The hardware configuration is compiled and saved
10	Download the configuration to the IM 151-7 CPU by means of the MPI and close the Hardware Editor.	The configuration is downloaded and a symbol for the IM 151-7 CPU appears in the right-hand part of the SIMATIC Manager window.

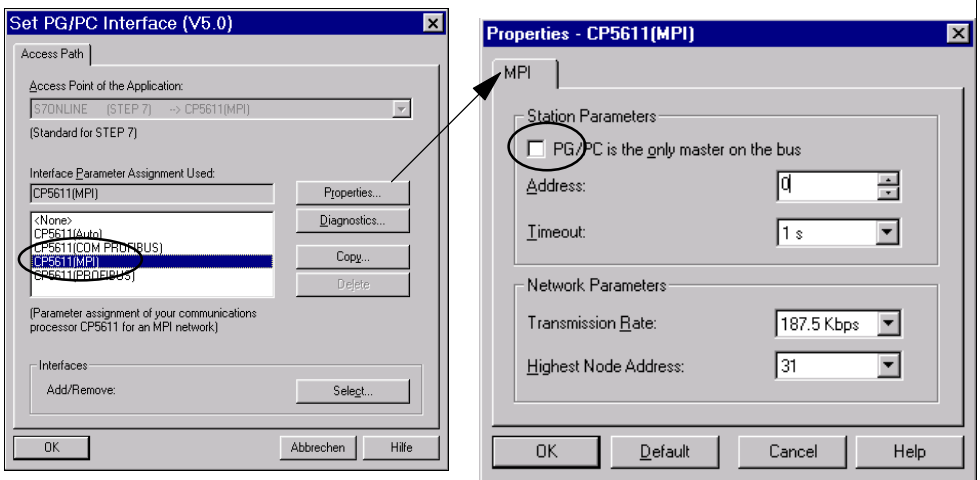
2.5 5th step: Programming the IM 151-7 CPU

Se-quence	Action	Result
1	In the SIMATIC Manager, navigate through **IM151-7 CPU** and **S7 Program** to the **Blocks** container.	
2	Double-click on the **OB 1** symbol in the right-hand part of the window.	The LAD/FBD/STL Editor used to edit the OB 1 block opens.
3	In the LAD/FBD/STL Editor, select the **LAD** command in the **View** menu to switch over to the LAD programming language.	A current path is displayed in Network 1.
4	Click on the horizontal line of the current path.	The line is highlighted.
5	In the toolbar, click on the –||– symbol (NO contact) twice and then on the –( ) symbol (coil) once.	The symbols are inserted into the current path.
6	Click on the red question mark at the left-hand NO contact in the current path.	The NO contact is highlighted and a text input box with a cursor appears in place of the question mark.
7	Enter *E1.0* and press *Return*.	The left-hand NO contact is given the name *E1.0*.
8	Enter *E1.1* and press *Return*. Enter *A2.0* and press *Return*.	The right-hand NO contact is given the name *E1.1*. The coil is given the name *A2.0.*
9	Close the Editor and answer the Save prompt with **Yes**.	The Editor is closed and OB 1 is saved.

2.6 6th step: Test Run

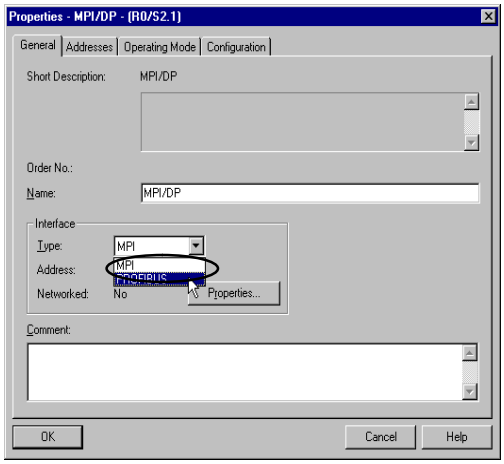
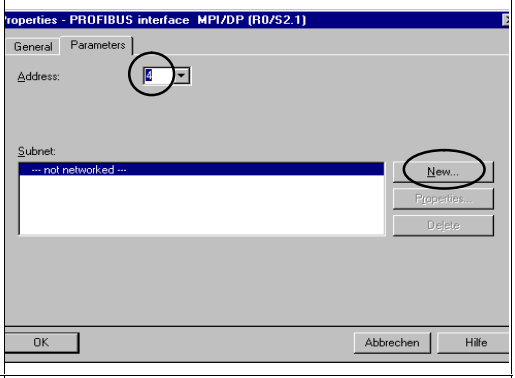
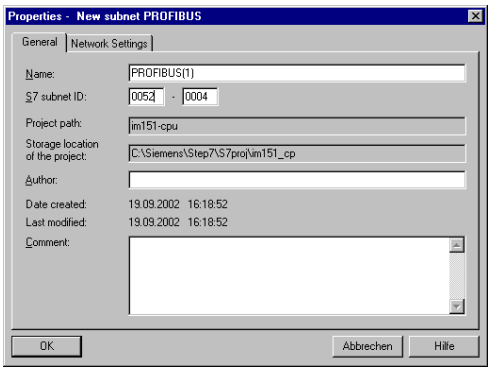
Se-quence	Action	Result
1	In the SIMATIC Manager, click on **Blocks** in the left-hand part of the window.	**Blocks** is highlighted.
2	Right-click in the right-hand part of the window and insert an empty organization block with the name *OB 82* into the block container. This block ensures that the S7-300-CPU is started even if the IM 151-7 CPU signals a diagnostic fault. Generate OB 86 in the same way. **Note:** OB 86 only becomes relevant in DP slave operation.	The blocks OB82 and OB86 appear next to the block OB 1.
3	Select the block container in the left-hand side of the window again. In the **PLC** menu, select the **Load** command to transfer the program and the hardware configuration to the IM 151-7 CPU. Confirm all windows that appear with **Yes**.	The program and configuration are downloaded from the PD to the IM 151-7 CPU.
4	Set the mode selector on the IM 151-7 CPU to *RUN*.	The *STOP* LED goes out. The *RUN* LED starts to flash and then remains lit.
5	Press the two buttons alternately.	The LEDs of inputs E1.0 and E1.2 light up alternately. The LED of output 2.0 does not light up.
6	Press the two buttons simultaneously.	The LEDs of inputs E1.0 and E1.2 (*1* and *5* LED of the DI) light up simultaneously. The LED of output 2.0 (*1* LED of the DO) lights up since the two buttons have been linked in the program by means of an AND function (= series connection) and assigned to output A2.0. A connected actuator or display element would then be activated.
7	Switch the mode selector of the IM 151-7 CPU to *STOP* and switch off the PS of the IM 151-7 CPU.	All LEDs go out.

2.7 7th step: Upgrading the IM 151-7 CPU as an I slave and commissioning the S7-300

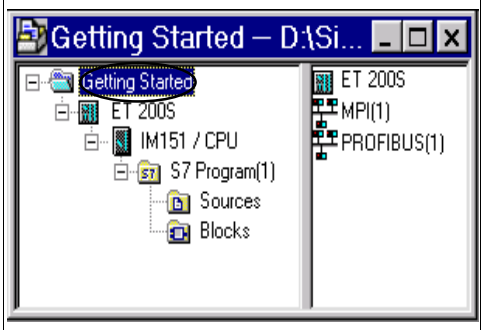
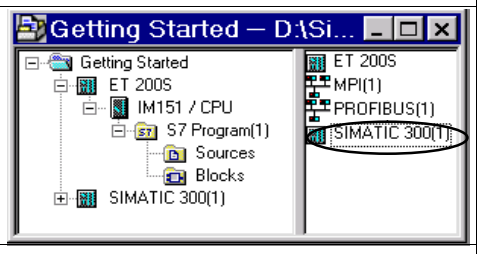
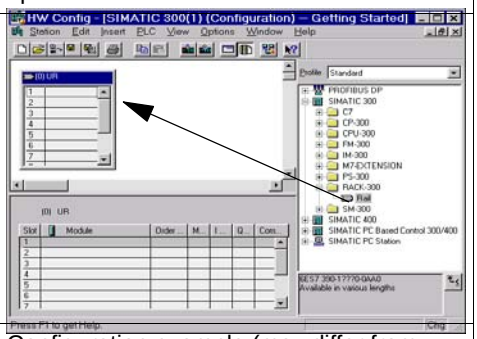
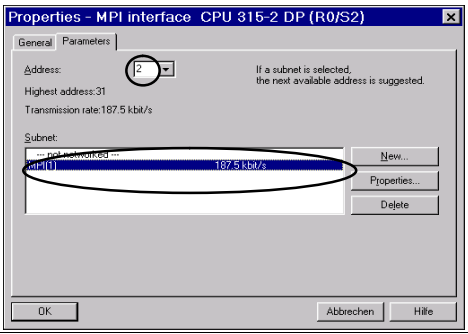
Se-quence	Description
1	Remove the connector of the PD cable from the IM 151-7 CPU.
2a	Start the Set PD/PC Interface program as described under step 3, point 4. Change the configuration of the PD/PC interface as follows:
2b	
3	Confirm the settings with OK and close the ”Set PD/PC Interface” program.

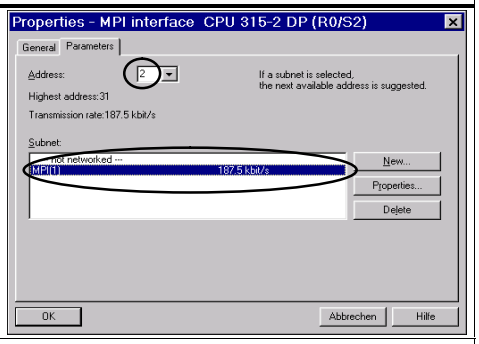
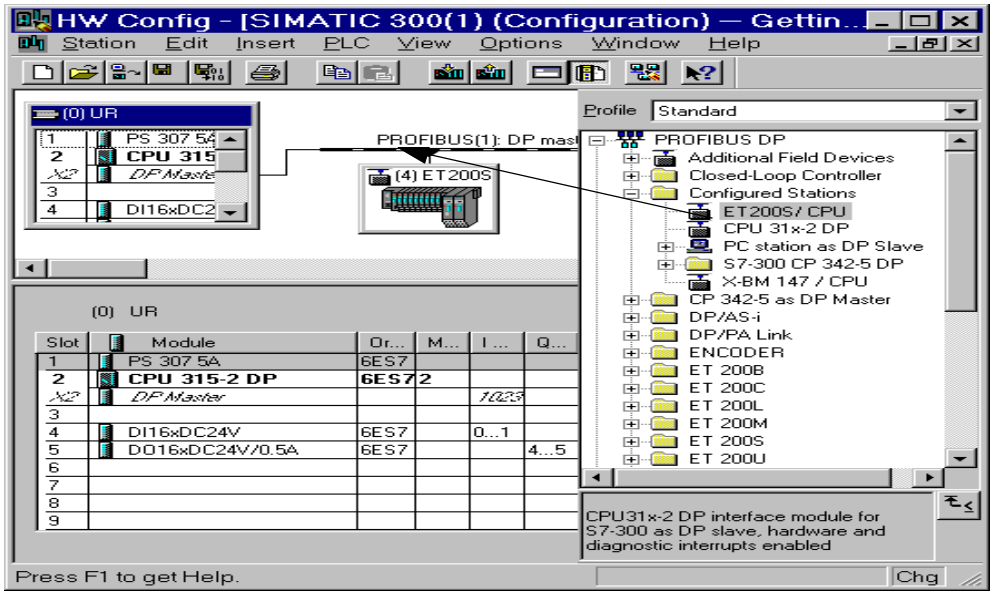
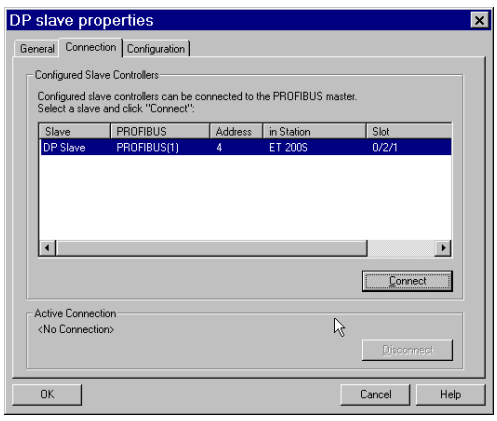
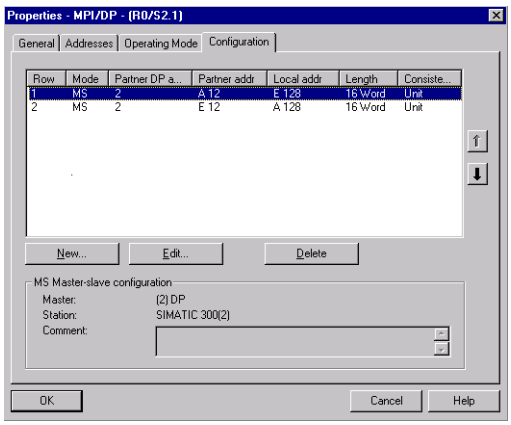
2.8 8th step: Configuring the IM 151-7 CPU as an I slave and the S7-300 as a DP master

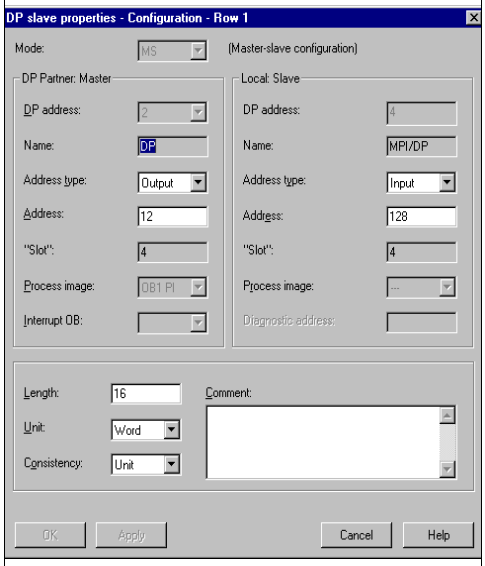
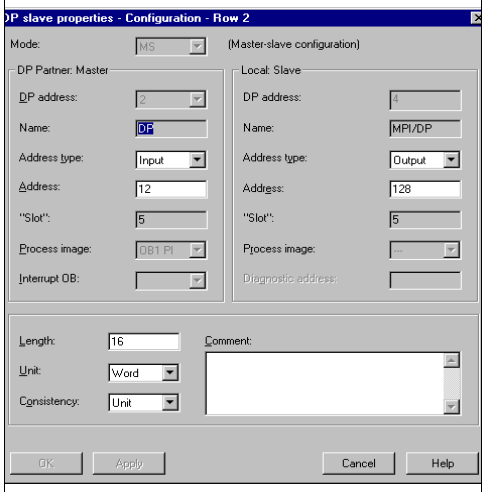
Change the configuration of the IM 151-7 CPU as follows:

Se-quence	Action	Result
1	Start the Hardware Configuration program for the IM 151-7 CPU as described in step 4.	The Editor used to edit the hardware configuration opens.
2	In the **Properties – MPI/DP** menu, select the interface type **PROFIBUS**.	
3	The **Properties – PROFIBUS Interface MPI/DP** window then opens. • Set the slave address to **4**. • Click on the **New** button; the **New Subnet PROFIBUS** window then opens.	
4	Check the settings in the **Properties – New Subnet PROFIBUS** window and confirm with **OK**.	
5	Select the command Save and compile in the **Station** menu.	The hardware configuration is compiled and saved.
6	Download the configuration to the IM 151-7 CPU by means of the MPI and close the Hardware Editor.	The ET 200S now has the DP address 4; the Editor is closed.

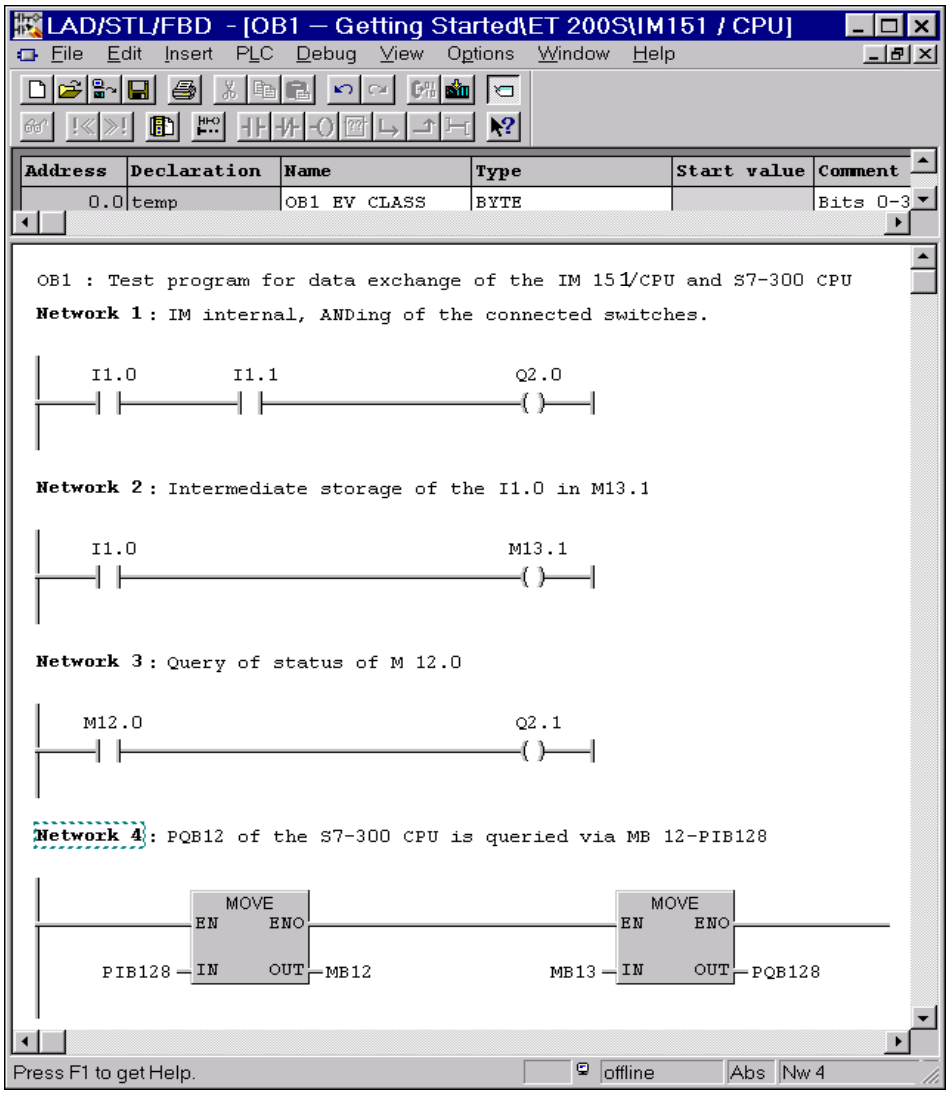
Configure the S7-300-CPU as follows:

Se-quence	Action	Result
1	In the SIMATIC Manager, select the project **Getting Started** in the left-hand part of the window.	
2	Insert the new S7-300 station into the project as described in step 4, point 3.	
3	In the SIMATIC Manager, click on the S7-300(1) station in the left-hand part of the window.	The **Hardware symbol appears in the right-hand part of the window.**
4	Double-click on the **Hardware** symbol in the right-hand part of the window.	The Editor used to edit the hardware opens.
5	If no catalog is shown in the right-hand part of the window, activate it by selecting the command **Catalog** in the **View** menu. In the catalog, navigate through **SIMATIC 300** to **Rack 300.** Drag and drop a mounting rail into the top left-hand window.	
6	Insert the PS whose order number matches the order number of your PS, at slot 1 as described in step 4. Repeat for the S7-300-CPU (slot 2), the S7-300-DI (slot 4) and the S7-300-DO (slot 5). **Note**: • A window appears when the S7-300-CPU is inserted. In this window, select ”PROFIBUS Network” and set the address to *2*. Confirm with **OK**.	Configuration example (may differ from your configuration): 

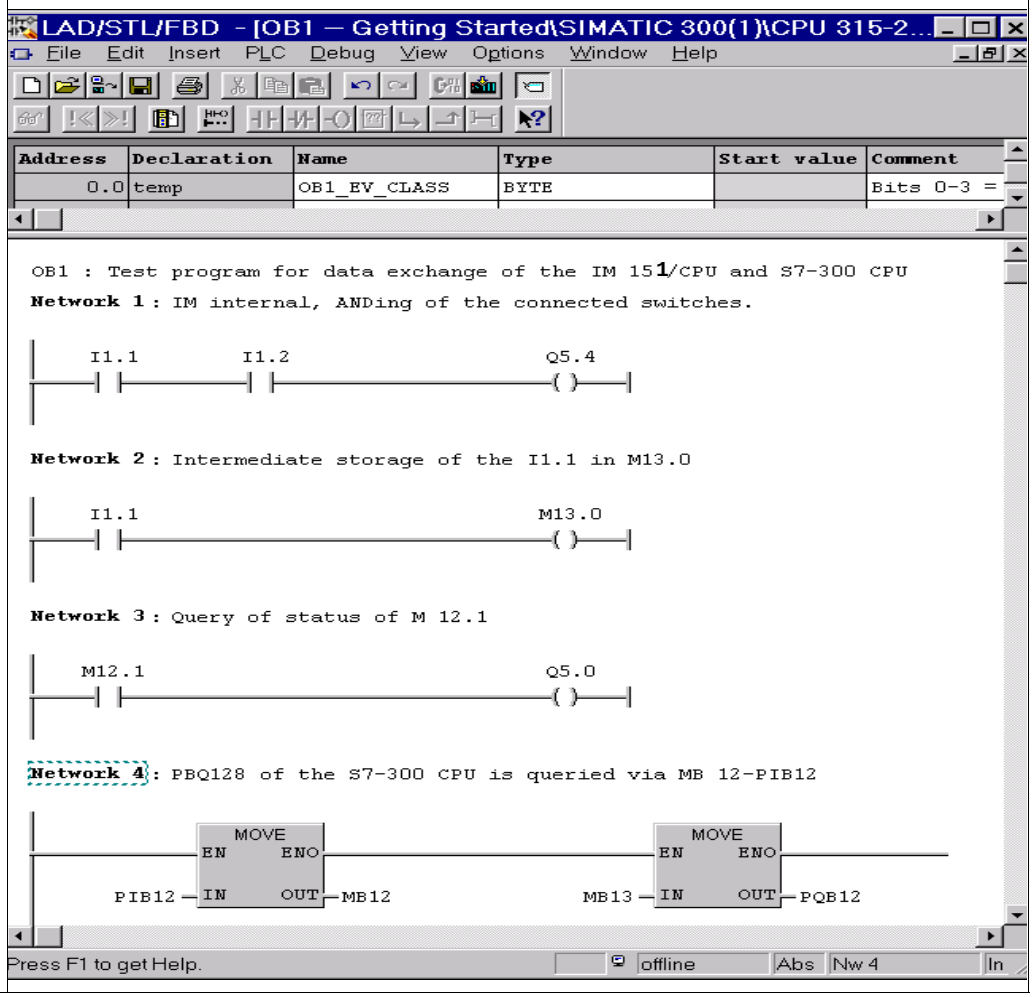
Se-quence	Action	Result
7	• In the bottom left-hand part of the Hardware Configuration Program window, double-click on **CPU 315-2 DP** (line **2**). • In the window that then appears, click on the **Properties** button on the **General** tab. • In the "MPI Network" window that then appears, check whether the address is set to *2*. If not, set the address. Confirm with **OK**.	
8	In the catalog, navigate through **PROFIBUS-DP** to **Configured Stations**. Drag and drop the **ET 200S/CPU** onto the **PROFIBUS Master System**.	
9	In the window that then appears, click on the **Interconnect** button. The **Properties – MPI/DP** window appears.	 

Se-quence	Action	Result
10	In the window from point 9, click on the **Edit** button and complete the form for line 1 as shown in the illustration. Then confirm with **OK**. Then click on the second line in the **Properties – MPI/DP** window and complete the form for line 2 as shown in the illustration. Then confirm with **OK**.	 
11	Select the command **Save and compile in the Station** menu.	The hardware configuration is compiled and saved.
12	Connect the PD to the MPI interface of the S7-300-CPU using a PD cable. Download the configuration to the CPU. Close the Hardware Editor.	The hardware configuration is loaded. The Editor is closed.

2.9 9th step: Programming the IM 151-7 CPU and the S7-300 CPU

Se-quence	Action	Result
1	In the SIMATIC Manager, navigate to the block container of the ET 200S. Double-click on the **OB1** symbol in the right-hand part of the window.	The LAD/FBD/STL Editor used to edit the OB 1 block opens.
2	Complete OB 1 of the IM 151-7 CPU as follows: 	
3	In the SIMATIC Manager, navigate to the block container of the S7-300. Double-click on the **OB1** symbol in the right-hand part of the window.	The LAD/FBD/STL Editor used to edit the OB 1 block opens.

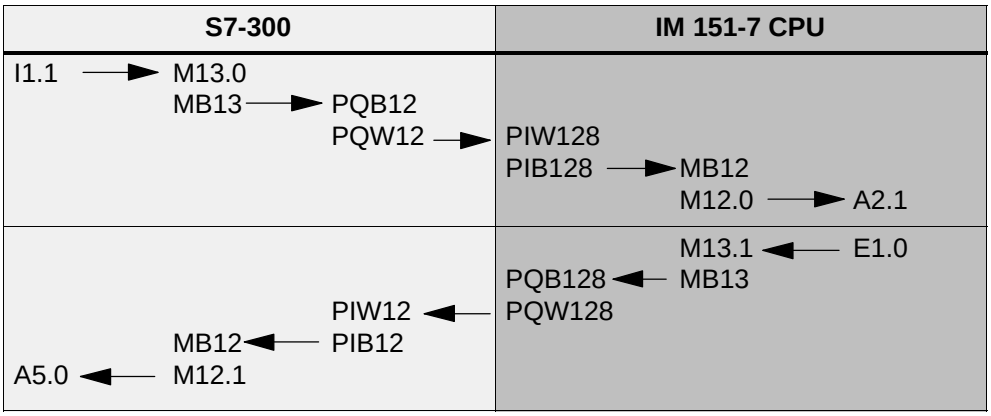
Se-quence	Action	Result
4	Complete the OB 1 of the S7-300 CPU as follows:	



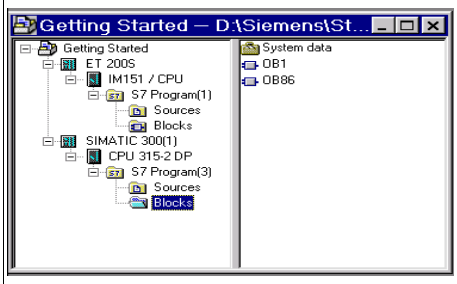
How it Works: The status of the button connected to E1.1 of the S7-300 is checked and stored temporarily in the memory marker M13.0. The entire memory byte MB13 is transferred to the peripheral output byte PQB12. In step 8 – Configuring the S7-300 (point 10) of the hardware configuration, you made settings to assign the area from PQW12 to PQW44 of the S7-300 CPU to the area from PIW128 to PIW160 of the IM 151-7 CPU.

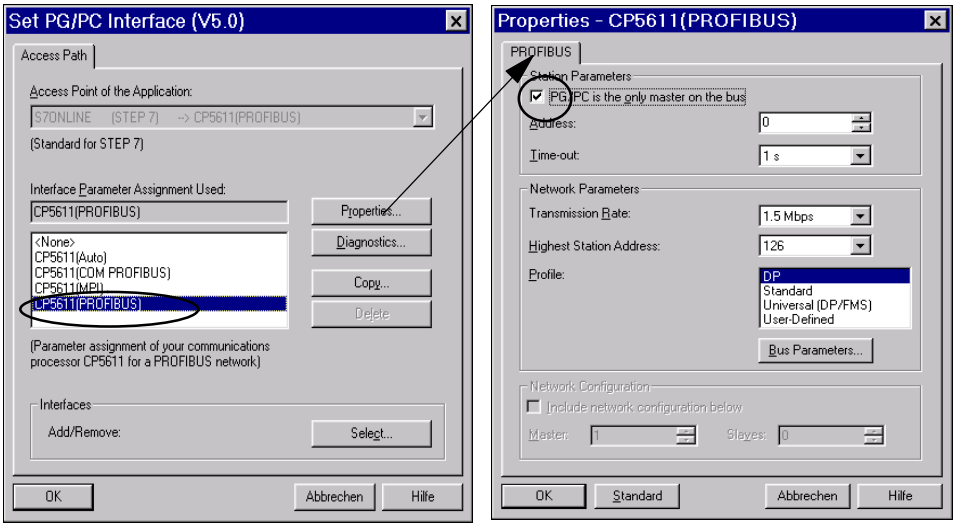
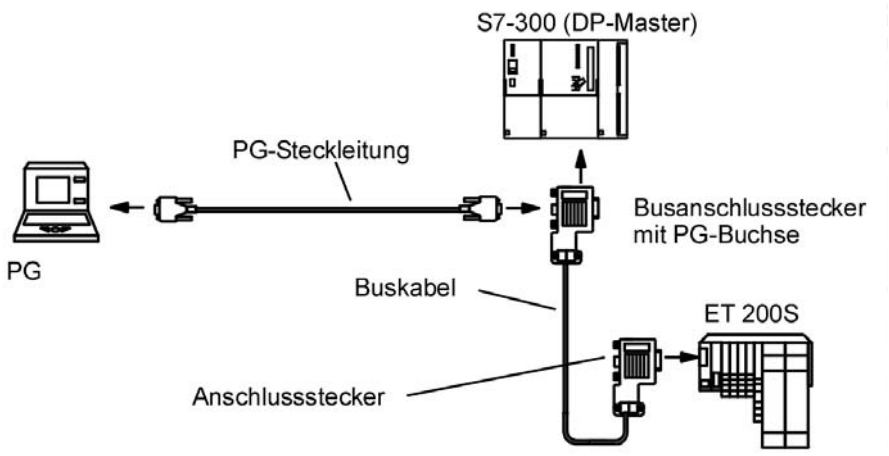
In the program of the IM, PIB128 is transferred into memory byte MB12. The memory marker M12.0 finally actuates the output A2.1.

This results in the following communication paths:



2.10 10th step: Commissioning and test run of the IM 151-7 CPU and S7-300

Se-quence	Action	Result
1	In the SIMATIC Manager, navigate to the block container of the S7-300 and insert an empty organization block with the name *OB 86* into the block container. This block ensures that the S7-300-CPU does not switch to STOP if a station of the IM 151-7 CPU fails/recovers. Generate OB 82 in the same way.	
2	Make sure that the mode selector of the S7 and IM is set to *STOP* Switch on the PS of the S7-300 and the PS of the ET 200S.	The IM 151-7 CPU and S7-300-CPU request a memory reset.
3	Reset the memory of the IM 151-7 CPU and S7-300-CPU as follows: • Press/rotate the mode selector to *MRES*. Hold the mode selector at this position until the *STOP* LED lights up for the second time and then remains lit (approx. 3 seconds). Then release the mode selector. • You must press/rotate the mode selector back to *MRES* within 3 seconds. The *STOP* LED begins to flash rapidly and the CPU performs a memory reset. You can now release the mode selector. The CPU has completed the memory reset when the *STOP* LED remains permanently lit again.	Both CPUs have undergone a memory reset.
4	In the **PLC** menu of the SIMATIC Manager, select the **Load** command to transfer the program and the hardware configuration to the S7-300-CPU. Confirm all windows that appear with **Yes**.	The program and configuration are downloaded from the PD to the CPU.

Se-quence	Action	Result
5	Start the **Set PD/PC Interface** program as described under step 3, point 4. Change the configuration of the PD/PC interface as follows: 	
6	Confirm the settings with OK and close the ”Set PD/PC Interface” program.	
7	Open the front panel of the S7-300-CPU. Connect the IM 151-7 CPU to the DP interface of the S7-300-CPU using a PROFIBUS-DP cable. Make sure that the terminating resistor is switched on at both connectors. Remove the connector of the PD cable from the MPI interface of the S7-300-CPU and connect it to the bus connector of the PROFIBUS-DP cable at the S7-300-CPU. Screw on the connector securely. Close the front panel of the S7-300-CPU if possible. 	
8	In the SIMATIC Manager, navigate to the block container of the ET 200S. Select the block container in the left-hand side of the window. In the **PLC** menu of the SIMATIC Manager, select the **Load** command to transfer the program and the hardware configuration to the IM 151-7 CPU. Confirm all windows that appear with **Yes**.	The program and configuration are downloaded from the PD to the IM 151-7 CPU.

Se-quence	Action	Result
9	Set the mode selector on the IM 151-7 CPU to *RUN*.	The *STOP* LED of the IM goes out. The *RUN* LED starts to flash and then remains lit. The *SF* LED lights up.
10	Set the mode selector on the S7-300-CPU to *RUN*.	The *STOP* LED of the S7 goes out. The *RUN* LED starts to flash and then remains lit. The *SF* LED of the IM goes out.
11	Press the two buttons on the S7-300 alternately.	The LEDs of the S7-300 inputs E1.1 and E1.2 light up alternately. The LED of output 5.4 does not light up.
12	Press the two buttons on the S7-300 simultaneously.	The LEDs of inputs E1.1 and E1.2 light up simultaneously. The LED of output 5.4 lights up since the two buttons have been linked in the program by means of an AND function (= series connection) and assigned to output A5.4.
13	Actuate the switch which is connected to E1.0 of the ET 200S.	The LEDs of the IM input E1.0 and the S7-300 output A5.0 light up.
14	Actuate the switch which is connected to E1.1 of the S7-300.	The LEDs of the S7-300 input E1.1 and the IM output A2.1 light up.

Diagnosis and troubleshooting

Faults which the CPU indicates with the group fault LED *SF* following memory reset can occur as a result of incorrect operation, incorrect wiring or incorrect hardware configuration.

How to diagnose such faults and messages is described in the following manuals:

- *Installation Manual S7-300;* Section 10.4
- *Programming with STEP 7 V5.x*; Chapter 21
- *Interface module IM 151-7 CPU*; Chapter 5

Other manuals

For more information on getting started we also recommend: *Getting Started: First Steps and Exercises With STEP 7 V5.x*.

All of these manuals can be downloaded free of charge from the Siemens homepage (Customer Support, Automation).

Addressing

3

Principles of Data Transfer Between the DP Master and the IM 151-7 CPU

This chapter contains information on the addressing of I/O modules and data transfer between the DP master and the IM 151-7 CPU.

The following alternatives are available for addressing the I/O modules:

- Slot-oriented address allocation:
  Slot-oriented address allocation is the default form of addressing, in which STEP 7 allocates a fixed module base address to each slot number.
- User-oriented address allocation:
  You can allocate each module any address within the available IM 151-7 CPU address area.

For information on the addressing of the IM 151-7 CPU on the PROFIBUS-DP, see Section 4.3.

Chapter Overview

3.1 Slot-Oriented Addressing of the I/O Modules

Slot-Oriented Address Allocation

In slot-oriented addressing (default addressing) each slot number in a module is allocated an address area in the IM 151-7 CPU.

Depending on the type of I/O module, the addresses are digital or analog (see Table 3-1). The address allocation is not fixed and can be changed, but there is a default address area.

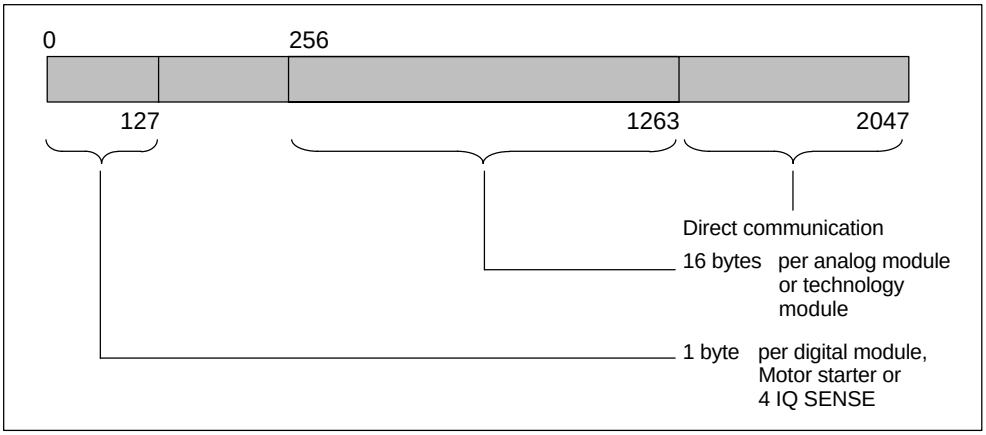


Figure 3-1 Structure of the default address area

Slot Assignment

The figure below shows an ET 200S configuration with digital electronic modules, analog electronic modules, process-related modules and the slot assignment.

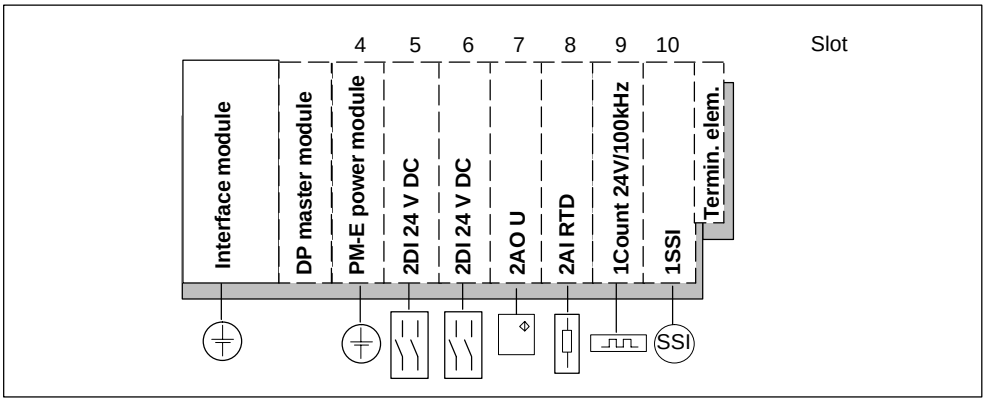


Figure 3-2 Slots on the ET 200S

Address Assignment

Depending on the slot, 1 byte is reserved for digital I/Os and 16 bytes are reserved for analog I/Os in the address areas of the IM 151/7 CPU for each I/O module (maximum of 63).

The table below indicates the default address assignment for analog and digital modules per slot. The address areas of the I/O modules are ”visible” only to an IM 151-7 CPU in the ET 200S, not to the associated DP master. The DP master has no direct access to the I/O modules.

Table 3-1 Addresses of the ET 200S I/O modules

Reserved Address Area	Slot Number									
	1	2	3	4	5	6	7	8	...	66
Digital modules, motor starters	IM 151-7 CPU*			-	1	2	3	4	...	62
Analog modules, process-related modules				-	272 to 287	288 to 303	304 to 319	320 to 335	...	1248 to 1263
Power modules				256	272	288	304	320		1248

* With X1 as the MPI-/DP interface and X2 as the DP master interface

The unassigned addresses in the range 64 to 127 are in the process image in default addressing and can be used any way you choose in the user program. If 2 bits in a byte are already occupied by a digital module, the remaining 6 bits cannot be used as you choose (e.g. the bits 1.4 to 1.7 in Figure 3-3).

You can use the bytes in the address areas that are not used by modules in any way you choose in your user program. In the configuration in Figure 3-3, for example, bytes 2 and 3 can be used as you choose.

Example of Slot-Oriented Address Assignment for I/O Modules

The figure below illustrates a sample ET 200S configuration, showing an example of the address allocation for I/O modules. The addresses for the I/O modules are predefined in default addressing.

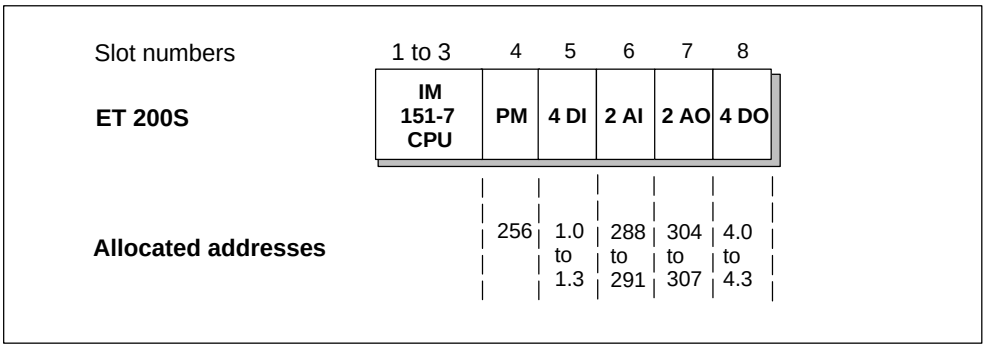


Figure 3-3 Example of address assignment for I/O modules

3.2 User-oriented addressing of the I/O Modules

User-oriented address allocation

User-oriented address allocation means you can select the following in units of 1 byte and independent of one another within the range 0 to 2047:

- Input addresses of modules
- Output addresses of modules

The addresses 0 to 127 are in the process image. Assign the addresses in *STEP 7*. When you do this, you define the base address of the module, on which all the addresses of the module depend.

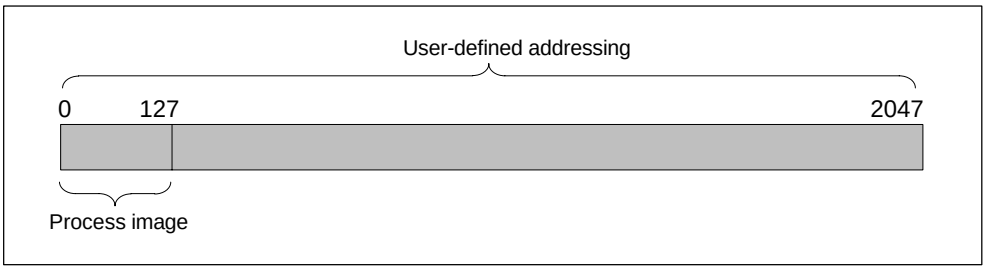


Figure 3-4 Structure of the address area for user-oriented addressing

Note

Bit-specific addressing is not possible in user-defined address allocation, and compression of digital channels is therefore not supported. It is not possible to compress addresses.

Advantages

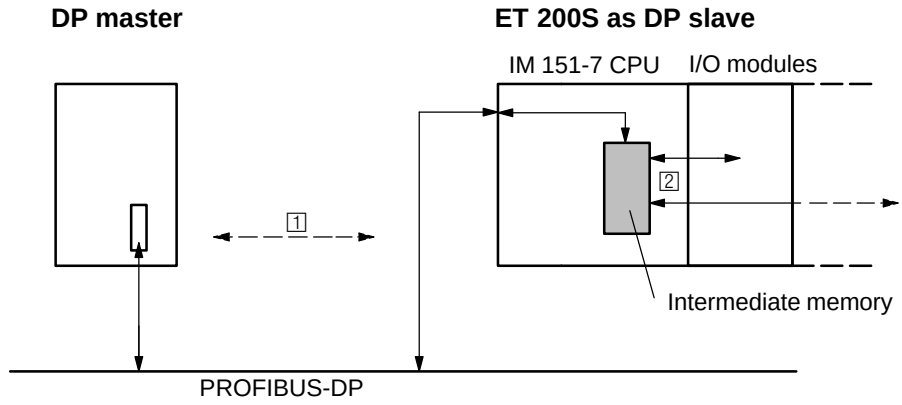
Advantages of user-defined address allocation:

- Optimum utilization of the address areas available, since ”address gaps” between the modules do not occur.
- When creating standard software, you can specify addresses that are independent of the configuration of the ET 200S station.

3.3 Data interchanger with the DP Master

User data transfer via an intermediate memory

The user data is located in an intermediate memory in the IM 151-7 CPU. This intermediate memory is always used when user data is transferred between the IM 151-7 CPU and the DP master. The intermediate memory consists of a maximum of 32 address areas.



1 Data transfer between the DP master and the ET 200S via an intermediate memory in the IM 151-7 CPU

2 Data transfer between the IM 151-7 CPU and I/O modules

Figure 3-5 Principles of data transfer between the DP master and the ET 200S with the IM 151-7 CPU

Address areas for user data transfer with the DP master

The ET 200S provides the PROFIBUS-DP with a maximum of 244 bytes of input data and 244 bytes of output data. This data can be addressed in the intermediate memory of the IM 151-7 CPU in up to 32 address areas.

An address area contains a maximum of 32 bytes. A maximum of 244 bytes is available for input and output data.

The address area begins at 128 by default. The data is entered with ”address gaps” starting at address 128.

Data consistency

You define data consistency as byte, word, or overall consistency per address area. Consistency can amount to up to 32 bytes/16 words per address area.

DP diagnostic address in *STEP 7*

When the ET 200S is configured with *STEP 7*, two diagnostic addresses are set. The ET 200S receives information on the status of the DP master or on a bus interruption by means of these diagnostic addresses (see Section 7.5). In DP slave mode, the diagnostic addresses are at 2045 and 2046 by default.

2045: Address for slot 2 (IM 151-7 CPU)
2046: Diagnostic address

Detailed information can be found in the *Online Help for STEP 7* under *Slot Model for I Slaves*.

Access to free areas in the process image

If you access available but unconfigured process image areas, no process image errors will be generated. You can therefore use inputs and outputs in the process image to which no I/O modules are allocated as markers.

3.4 Accessing the intermediate memory in the IM 151-7 CPU

Access in the user program

The following table tells you how to access the intermediate memory in the IM 151-7 CPU from the user program.

Table 3-2 Accessing the address areas

Access dependent on data consistency	The following applies
1-, 2- or 4-byte data consistency with load/transfer instructions	All areas parameterized with “unity” consistency can be accessed. You can address a maximum of 64 bytes of input data using load instructions and a maximum of 64 bytes of output data using transfer instructions (L PIB/PIW/PID; T PQB/PQW/PQD; see also *Instruction list*). The data consistency for word addressing is 2 bytes; for double-word addressing it is 4 bytes. Access is also possible via the process image.
1- to 32-byte data consistency on the PROFIBUS-DP with SFC 14 and SFC 15	If the address area of consistent data is in the process image, this area is updated automatically. If you want to access data in the intermediate memory, you have to read the input data with SFC 14 ”DPRD_DAT” and write the output data with SFC 15 ”DPWR_DAT”. These SFCs have data consistency of 1 to 32 bytes. You can only copy the input data read with SFC 14 as a block of 1 to 32 bytes to a memory marker address area, for example, where it can be addressed with A M x.y. You can also write only one block of 1 to 32 bytes as output data with SFC 15 (see also the *System and Standard Functions*) Reference Manual. If you access areas with “whole length” consistency, the length in the SFC must correspond to the length of the parameterized area. It is also possible to address the consistent areas directly (for example, L PIW or T PQW).

Rules for address allocation

You must comply with the following rules when allocating addresses for the ET 200S with the IM 151-7 CPU:

- Assignment of the address areas:
  - Input data for the ET 200S is **always** output data for the DP master
  - Output data for the ET 200S is **always** input data for the DP master
- You access the data in the user program using load/transfer instructions or SFCs 14 and 15.
- The length, unit and consistency of the associated address areas for the DP master and the DP slave must be identical.
- Addresses for the master and the slave can be different in the logically identical intermediate memory (mutually independent logical I/O address areas in the master and the slave CPU)

When the IM 151-7 CPU is configured with *STEP 7* for operation in the S5 or in non-Siemens systems, it is clear that only the logical addresses within the slave CPU are allocated. The addresses are then assigned in the master system using the specific configuration tool of the master system.

Addressing Interface in *STEP 7*

The following table illustrates the principles of address allocation. You will also find this table in the *STEP 7* interface. You must set the mode ”MS” (for master slave) or ”DX” (direct connection) in *STEP 7* (see Section 4.5).

Table 3-3 Addressing interface in *STEP 7 V5.1 (extract)*

	Mode	**Master**		**PROFIBUS-DP Partner**		**Parameters**		
		I/O	Address	I/O	Address	Length	Unit	Consistency
1	MS	Q	200	I	128	4	Byte	Unit
2	MS	Q	300	I	132	8	Byte	Total length
3	MS	I	700	Q	128	4	Word	Unit
4	MS	I	50	Q	136	4	Byte	Unit
	MS: Master Slave	Address areas in the DP master CPU		Address areas in the IM 151-7 CPU		These address area parameters must be identical for the DP master and the IM 151-7 CPU		

Sample Program

Below you will see a sample program for data interchange between the DP master and the DP slave.

You can find the addresses in Table 3-3.

SFCs 14 and 15 are called by specifying the logical address in hexadecimal format.

In the IM 151-7 CPU

```
Data preprocessing in the DP slave:

L       2                           Load actual value 2 and
T       MB      6                   transfer to memory byte 6.
L       IB      0                   Load input byte 0 and
T       MB      7                   transfer to memory byte 7.
```

```
Forward data to DP master

L       MW      6                   Load memory word 6 and
T       PQW     136                 transfer to peripheral output word 136
```

In the DP Master CPU

```
Postprocess received data in the DP master:

L       PIB     50                  Load peripheral input byte 50 and
T       MB      60                  transfer to memory byte 60.
L       PIB     51                  Load peripheral input byte 51 and
L       B#16#3                      load byte 3;
+       I                           add the values as integer data type and
T       MB      61                  transfer the result to memory byte 61.
```

```
Data preprocessing in the DP master:

L       10                          Load actual value 10 and
+       3                           add 3,
T       MB      67                  transfer the result to memory byte 67.
```

```
Send the data (memory bytes 60 to 67) to the DP slave:

CALL    SFC     15                  Call system function 15:
 LADDR:= W#16#12C                   Write the data to the output address area as of
 RECORD:= P#M60.0 Byte8             address 300 (12C hexadecimal) with a length of 8
 RET_VAL:=MW 22                     bytes as of memory byte 60.
```

In the IM 151-7 CPU

```
Receive data from the DP master (stored in MB 30 to 37):

CALL    SFC     14                  Call system function 14:
 LADDR:= W#16#84                    Write the data from the input address area as of
 RET_VAL:=MW 20                     address 132 (84 hexadecimal) with a length of 8
 RECORD:=P#M30.0 Byte8              bytes to memory byte 30.
```

```
Postprocess received data:

L       MB      30                  Load memory byte 30 and
L       MB      37                  load memory byte 37;
+       I                           add the values as integer data type and
T       MW      100                 transfer the result to memory byte 100.
```

User Data Transfer in STOP Mode

The user data in the intermediate memory is processed differently depending on whether the DP master or the DP slave (IM 151-7 CPU) goes into STOP mode.

- If the IM 151-7 CPU goes into STOP mode: The data in the intermediate memory (outputs only from the slave's viewpoint) of the IM 151-7 CPU are overwritten with "0"; i.e. the DP master or a recipient in direct communication reads "0".
- If the DP master goes into STOP mode: The current data in the intermediate memory of the IM 151-7 CPU (inputs in the slave, outputs in the master) are retained and can be read out in the user program of the IM 151-7 CPU.

S5 DP master

If you use the IM 308–C as a DP master and the IM 151–7 CPU as an I slave, the following applies to the exchange of consistent data:
You must program FB 192 in the IM 308–C to enable the transfer of consistent data between the DP master and the I slave. The effect of FB 192 is that the data of the IM 151–7 CPU is only output or read out continuously in a single block.

ET 200S in the PROFIBUS Network

4

Introduction

You can integrate the ET 200S with the IM 151-7 CPU as a node in a PROFIBUS network. This chapter contains a description of a typical network configuration with the IM 151-7 CPU. It also tells you which functions can be executed via the PD or OP on the ET 200S and which options are available for direct connection. The available communication utilities can be found in Section 8.8.

Equidistance

As of *STEP 7* V5.1 + SP4, you can parameterize bus cycles of the same length (equidistant) for PROFIBUS subnets with IM 151-7 CPU. You will find a detailed description of the functions in the *Online Help for STEP 7*.

DP master functionality

In combination with the DP master module, the IM 151-7 CPU can be used as a DP master. Further information can be found in Section 6.

Chapter overview

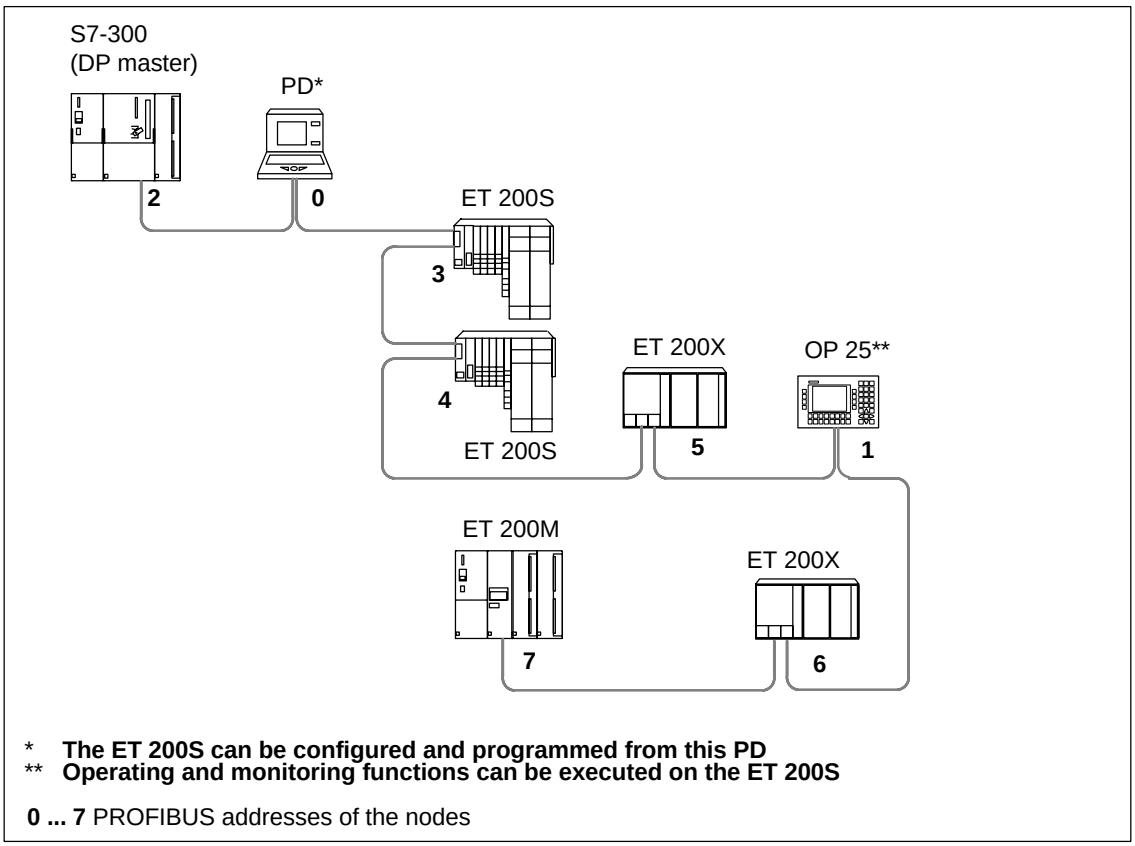
More information

You will find more information on the structure of networks in the manual for the DP master.

4.1 ET 200S in the PROFIBUS network

Structure of a PROFIBUS network

The figure below illustrates the basic structure of a PROFIBUS network with one DP master and several DP slaves.



* **The ET 200S can be configured and programmed from this PD**
** **Operating and monitoring functions can be executed on the ET 200S**

0 ... 7 PROFIBUS addresses of the nodes

Figure 4-1 Example of a PROFIBUS network

Hardware prerequisites in the PD/OP for accessing the ET 200S

Before you can access an IM 151-7 CPU from a PD/OP, the PD/OP must fulfill the following requirements:

- it must have an integrated PROFIBUS-DP interface or DP card; or
- it must have an integrated MPI interface or MPI card.

Data transfer rates over 1.5 MBaud require an active connecting cable for the PD connection.

Access to the ET 200S

The IM 151-7 CPU is a passive/active bus node. The programs and configuration of the IM 151-7 CPU can be transferred to the IM 151-7 CPU by choosing ”Load PLC” from the PD in SIMATIC Manager. All the other diagnostic and test functions are also possible with the PD.

If the PD is currently the only active bus node, this must be set beforehand in SIMATIC Manager by choosing the ”Set PD/PC Interface” menu command (see Section 4.4).

However, you can still install OPs/OSs (operator panels/operator stations) as fixed components of the PROFIBUS network for operating and monitoring functions.

You cannot access an ET 200S from more than 12 devices in parallel:

- 1 connection is reserved for the PD.
- 1 connection is reserved for an OP or an OS.
- 10 connections are available as desired for PDs, OPs/OSs and CPUs

We recommend that you allocate a PROFIBUS address to the PD/OP in the same way as for other network nodes (see Figure 4-1).

Active/passive DP interface of the IM 151-7 CPU

You set the mode of the DP interface at the IM 151-7 CPU during configuration in the **Properties – MPI/DP** window:

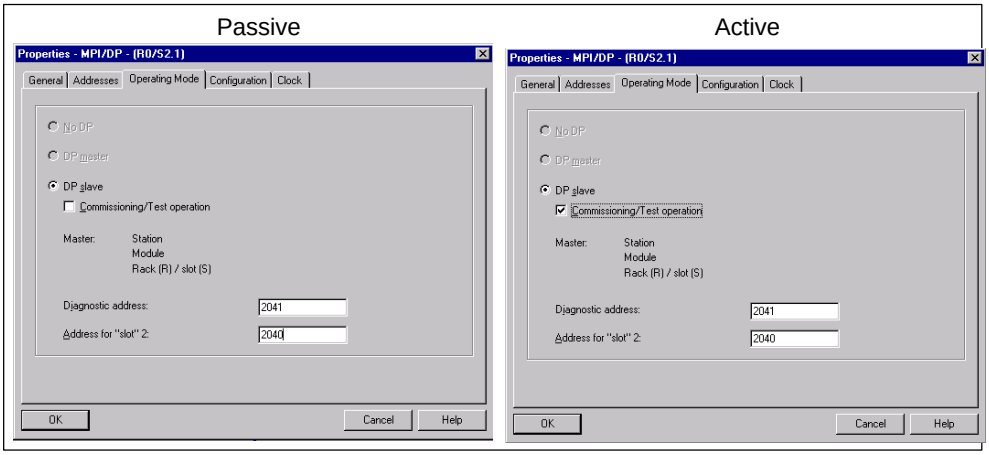


Figure 4-2 Setting the mode of the DP interface at the IM 151-7 CPU

Depending on the DP interface setting, the IM 151-7 CPU will behave in the following way:

Table 4-1 Behavior of the IM 151-7 CPU depending on the DP interface setting

	DP Interface of the IM 151-7 CPU	
	Passive	**Active**
Transmission rate detection	Yes	No
Testing and commissioning functions	Slow	Fast
Bus cycle time	Fast	Slow
Diagnosis via BF LED	See Section 7.4	
Routing (with plugged-in DM master module)	No	Yes

Maximum data transfer rate and cable length with a PD connecting cable

You can obtain a maximum data transfer rate of 1.5 Mbaud using the PD connecting cable. The cable length may not exceed 3 meters.
The PD connecting cable should only be connected for an extended period of time during startup and service.

Data transfer rates over 1.5 MBaud require an active connecting cable for the PD connection (order number: 6ES7 901-4BD00-0XA0).

Examples for the connection of the PD/OP on the ET 200S

- The PD/OP is connected to the PROFIBUS-DP interface of the DP master, but can be connected just as well to any other station in the DP network, including the ET 200S.

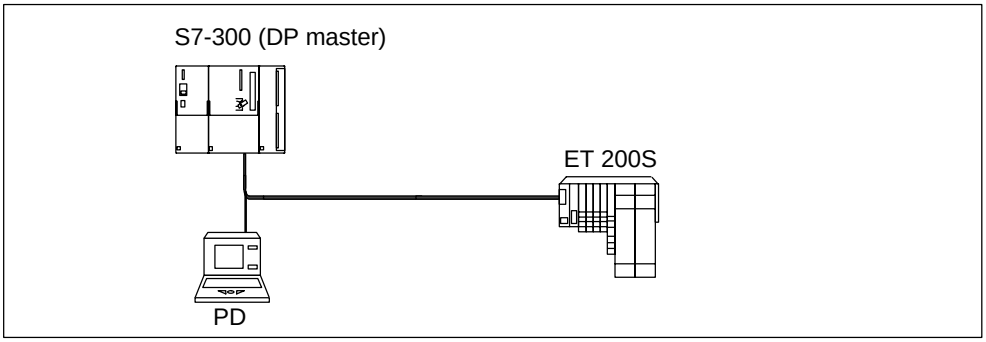


Figure 4-3 The PD/OP accesses the ET 200S via the DP interface in the DP master

- The PD is directly connected to the ET 200S (you don't add the ET 200S to the PROFIBUS network until later).
  Note: Depending on the DP interface (active/passive), a special setting is required in *STEP 7* (see Section 4.4).

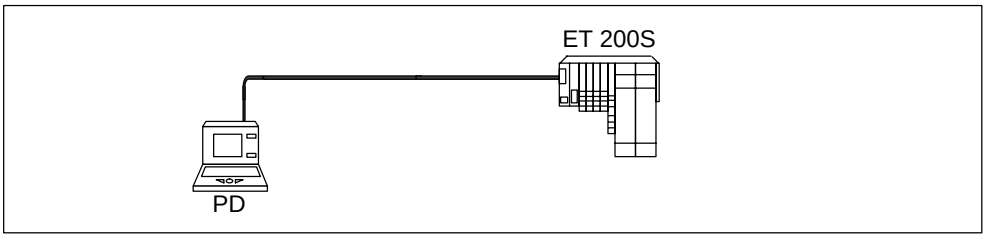


Figure 4-4 The PD directly accesses the ET 200S

- The PD can also be a direct DP node, although a spur line (e.g. PD connecting cable) is not permissible with a transmission rate greater than 1.5 Mbaud. This requires an active spur line.

4.2 Network components

To connect the ET 200S to the PROFIBUS-DP network, you need the following network components:

Table 4-2 Network components

Purpose	Network components	Order numbers
To set up the network	Cables (e.g. 2-core, shielded or 5-core, unprepared)	6XV1 830-0AH10 (2-core) 6XV1 830-0BH10 (2-core with PE sheath) 6XV1 830-3CH10 (2-core, for festoon attachment) 6XV1 830-3BH10 (drum cable) 6XV1 830-3AH10 (direct-buried cable) 6ES7 194-1LY00-0AA0-Z (5-core with PVC sheath) 6ES7 194-1LY10-0AA0-Z (5-core; oil-resistant, can be dragged, conditionally resistant to welding; with PUR sheath)
To connect the PD and the ET 200S on the PROFIBUS-DP network	Bus connector without a PD socket (up to 12 Mbaud)	6ES7 972-0BA10-0XA0 (with a straight outgoing cable unit) 6ES7 972-0BA40-0XA0 (with a slanted outgoing cable unit)
To make a dual connection – for the PD and the DP master on the PROFIBUS-DP network, for example – via a DP interface (see Figure 4-5)	Bus connector with a PD socket (up to 12 Mbaud)	6ES7 972-0BB10-0XA0 (with a straight outgoing cable unit) 6ES7 972-0BB40-0XA0 (with a slanted outgoing cable unit)
To connect the PD to the bus connector with the PD socket	PD connecting cable (up to 1.5 Mbaud)	6ES7 901-4BD00-0XA0

Example of the use of network components

The figure below shows the example from Figure 4-3 with the use of the network components. Connecting the bus cable to the bus connector is described in the Product Information document for the bus connector.

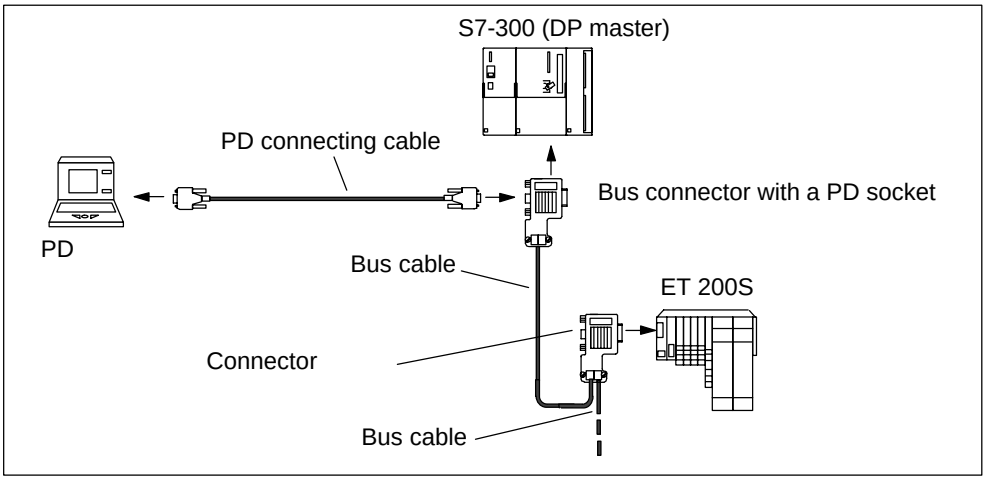


Figure 4-5 Connecting the DP network

4.3 PROFIBUS address

Features

Use the PROFIBUS address to specify the address at which the IM 151-7 CPU is contacted on the PROFIBUS-DP.

Prerequisites

- The permitted PROFIBUS-DP addresses are 1 to 125.
- Each address can be allocated only once on the PROFIBUS-DP.

Startup without DP configuration on the Micro Memory Card (MMC) (initial startup)

Following POWER ON, the coexistent interface on the IM 151-7 CPU powers up as an MPI interface with the address 2, HSA 31 and 187.5 kBaud. The I slave functionality of the IM 151-7 CPU is not yet available. All PD functions listed in Section 4.4 are possible with the interface.

Several ET 200S units with IM 151-7 CPUs as I slaves on one PROFIBUS network must be commissioned step by step. After each individual IM 151-7 CPU has been switched on, STEP 7 must be used to transfer a configuration with DP address to the IM 151-7 CPU.

Note

The bus parameters are retentive, i.e.bus parameters that have been configured (e.g. address, transmission rate) are retained

- with POWER OFF
- if there is no longer a configuration on the IM 151-7 CPU (e.g. after SDBs have been deleted or following POWER ON without MMC)

Startup with DP configuration on the Micro Memory Card (MMC)

As soon as a DP configuration has been downloaded to the IM 151-7 CPU, the data stored on the MMC is used on startup.

Following POWER ON, the IM 151-7 CPU as the I slave powers up with the configured address and waits for parameter assignment by the DP master.

As active PROFIBUS node, the IM 151-7 CPU adopts the configured transmission rate.
As passive PROFIBUS node, the IM 151-7 CPU searches for the transmission rate.

4.4 Functions via the PD/OP

You can use the PD to:

- Configure the IM 151-7 CPU with ET 200S modules and put them into operation on the PROFIBUS-DP
- Program the IM 151-7 CPU
- Execute test functions such as ”Monitor/Modify Variables” and ”Program Status”

  Execute commissioning functions such as ”Start” and ”Memory Reset”
- Display the module status (i.e. for the IM 151-7 CPU, for example, you can display the utilization of the load and working memory, stack contents and diagnostic buffer contents)

You can use the OP to:

- Operate and monitor

You will find a detailed description of the functions in the online help for *STEP 7*.

Running the IM 151-7 CPU as a passive I slave on the PD – required settings in *STEP 7*

If you connect an IM 151-7 CPU directly to a PD, you must set the PD interface in *STEP 7* to allow communication between the two partners. Proceed as follows:

1. In *STEP 7*, choose the ”Set PD/PC Interface” tool (**Start > STEP 7 > Set PD/PC Interface**).
2. Set the interface of your PD to PROFIBUS.
3. Call the properties of the PROFIBUS network.
4. Set the properties so that the PD/PC is the only active master on the bus.

If you subsequently configure a DP master for the network and want to go online, you should cancel these settings; additional security functions are thus activated against bus faults.

Force test function

In the case of the IM 151-7 CPU, you can preset the inputs and the outputs in the process image with fixed values using the ”Force” function.

The values (force values) you have preset can still be controlled in the IM 151-7 CPU by the user program and by PD/OP functions. This is shown in Figure 4-6.

You can force a maximum of 10 variables with the IM 151-7 CPU.

Caution

The force values in the process-image input table can be overwritten by write commands (for example T IB x, = I x.y, copy with SFC, etc.) as well as by I/O read commands (L PIW x, for example) in the user program or by PD/OP write functions.

Outputs preset with force values only return the force value provided the user program does not execute any write accesses to the outputs using I/O write commands (e.g. T PQB x) and provided no PD/OP functions write to these outputs.

It is important to note that force values in the process-image input/output table cannot be overwritten by the user program or by PD/OP functions.

Principle behind forcing with the IM 151-7 CPU

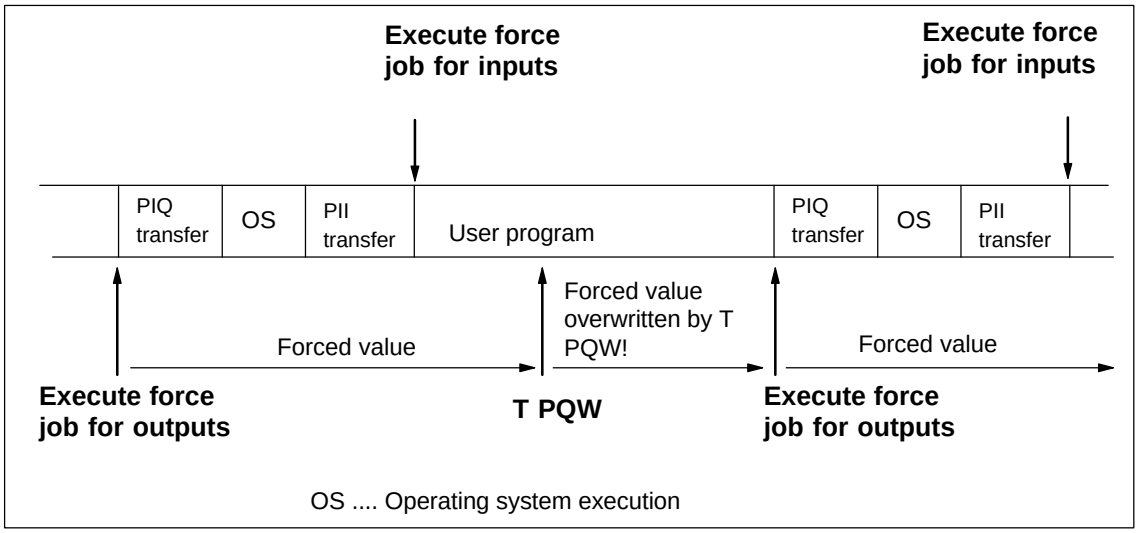


Figure 4-6 Principle behind forcing

Application example

Prerequisite: There is no direct I/O access in your user program.

If, for example, an enable sensor (f) in your system is defective and it continually indicates a logical 0 to your user program, for example, via input 1.2, you can bridge this sensor by forcing the input to 1, ensuring that your system continues to operate.

Warning

However, because the sensor is out of operation, you must monitor the functionality by different means to avoid injury to the operator and damage to the machine.

4.5 Direct communication

You can configure the IM 151-7 CPU as an intelligent slave with *STEP 7* V5.1 for direct communication. Direct communication is a special communication relationship between PROFIBUS-DP nodes.

Principle

Direct communication is characterized by the fact that the PROFIBUS-DP nodes ”listen in” to find out which data a DP slave is sending back to its DP master. Using this function, the eavesdropper (recipient) can directly access changes to the input data of remote DP slaves.

During configuration in *STEP 7*, you set via the relevant I/O input addresses the address area of the recipient at which the required data of the sender is to be read.

Example

Figure 4-7 gives you an example of the relationships you can configure for direct communication in *STEP 7* with an IM 151-7 CPU. Other DP slaves can only be senders here.

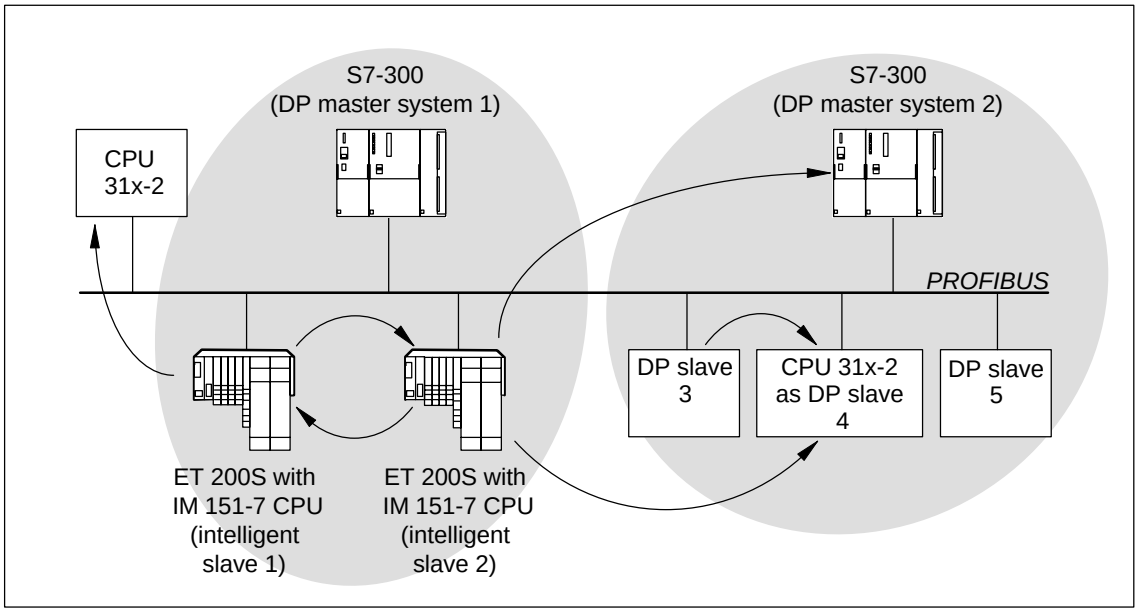


Figure 4-7 Direct communication with the IM 151-7 CPU

Functionality in direct communication

The IM 151-7 CPU offers the following functionality in direct communication:

- Transmitter:
  As an I slave, the IM 151-7 CPU sends the process outputs, configured for direct communication, to all bus nodes as a broadcast frame. Other recipients filter the relevant data from this broadcast frame.
- Receiver:
  Filtering of the data from the broadcast frame sent by transmitters which have been configured using *STEP 7* as being relevant for direct communication.

Diagnostics in direct communication

Only the results of connection monitoring can be used in the diagnostics of the DP slaves configured for direct communication, because diagnostic messages of the DP slaves that have been listened in on are only reported to the DP master.

The asynchronous OB 86 is called in the event of station failure and reintegration. If data is accessed during a station failure of the sender, an I/O access error is detected and OB 122 is called. Only the identifiers ”module plugged” and ”module available” are relevant for the module status data.

ET 200S in the MPI Network

5

Introduction

You can integrate the ET 200S with the IM 151-7 CPU as a node in an MPI network. This chapter contains a description of a typical network configuration with the IM 151-7 CPU. Section 4.4 describes which functions can be executed on the IM 151-7 CPU using a PD or OP. The available communication utilities can be found in Section 8.8.

Information on clock synchronization via the MPI interface is found in the *STEP 7 Online Help*.

Chapter overview

5.1 ET 200S in the MPI network

Structure of an MPI network

The figure below shows an example of an MPI network configuration.

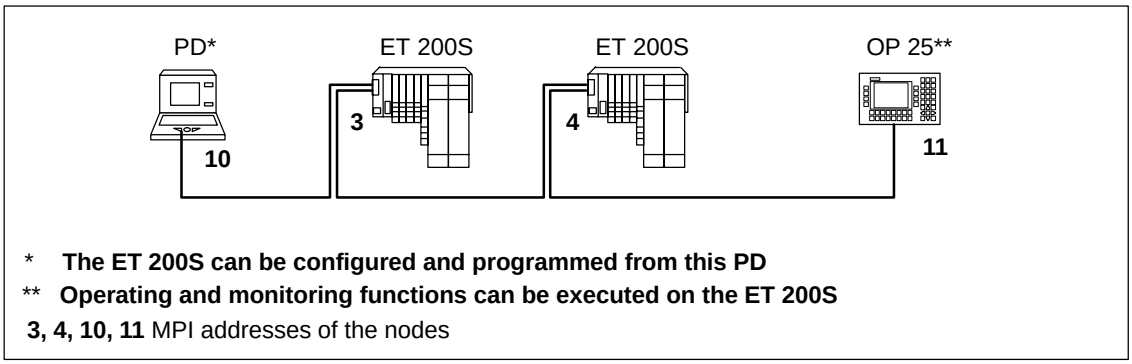


Figure 5-1 Example of an MPI network

Hardware prerequisites in the PD/OP for accessing the ET 200S

Before you can access an IM 151-7 CPU from a PD/OP, the PD/OP must fulfill the following requirements:

- it must have an integrated MPI interface or MPI card or
- it must have an integrated PROFIBUS-DP interface or DP card.

Transmission rates

In the MPI network, all MPI transmission rates are possible with the IM 151-7 CPU.

Network components

You configure an MPI network using the same network components as those used for a PROFIBUS-DP network (see Section 4.2).

Maximum data transfer rate and cable length with a PD connecting cable

You can obtain a maximum data transfer rate of 1.5 Mbaud using the PD connecting cable. The cable length may not exceed 3 meters.
The PD connecting cable should only be connected for an extended period of time during startup and service.

Data transfer rates over 1.5 MBaud require an active connecting cable for the PD connection (order number: 6ES7 901-4BD00-0XA0).

5.2 MPI address

Features

With the MPI address, you determine the address under which the IM 151-7 CPU is accessed in the MPI network.

Prerequisites

- The permitted MPI addresses are 0 to 126.
- Each address can be allocated only once on the MPI network.

Recommendations for MPI addresses

- Assign MPI addresses greater than ”2” to the fixed nodes in the MPI network.
- Reserve the MPI address ”0” for a service PD and ”1” for a service OP which, if necessary, can be connected to the MPI network at short notice.
- Reserve the MPI address ”2” for a CPU. This prevents double MPI addresses occurring when a CPU with default settings is installed in the MPI network (e.g. when a CPU is exchanged).

Startup without configuration on the Micro Memory Card (MMC) (initial startup)

Following POWER ON, the coexistent interface on the IM 151-7 CPU powers up as an MPI interface with the address 2, HSA 31 and 187.5 kBaud. All PD functions listed in Section 4.4 are possible with the interface.

Note

The bus parameters are retentive, i.e. bus parameters that have been configured (e.g. address, transmission rate) are retained

- with POWER OFF
- if there is no longer a configuration on the IM 151-7 CPU (e.g. after SDBs have been deleted or following POWER ON without MMC)

Startup with configuration on the Micro Memory Card (MMC)

As soon as a configuration has been downloaded to the IM 151-7 CPU, the data stored on the MMC is used on startup.

DP Master Module

6

Introduction

In combination with the DP master module, you can operate the IM 151-7 CPU as a DP master. In this case, the IM 151-7 CPU can be

- integrated in a PROFIBUS network as an I slave or
- operated in stand-alone mode (MPI).

You will require *STEP 7* of V5.2 + SP1 or higher to configure the DP master functionality. The following figure shows an example of a network structure in which the IM 151-7 CPU acts as a DP master.

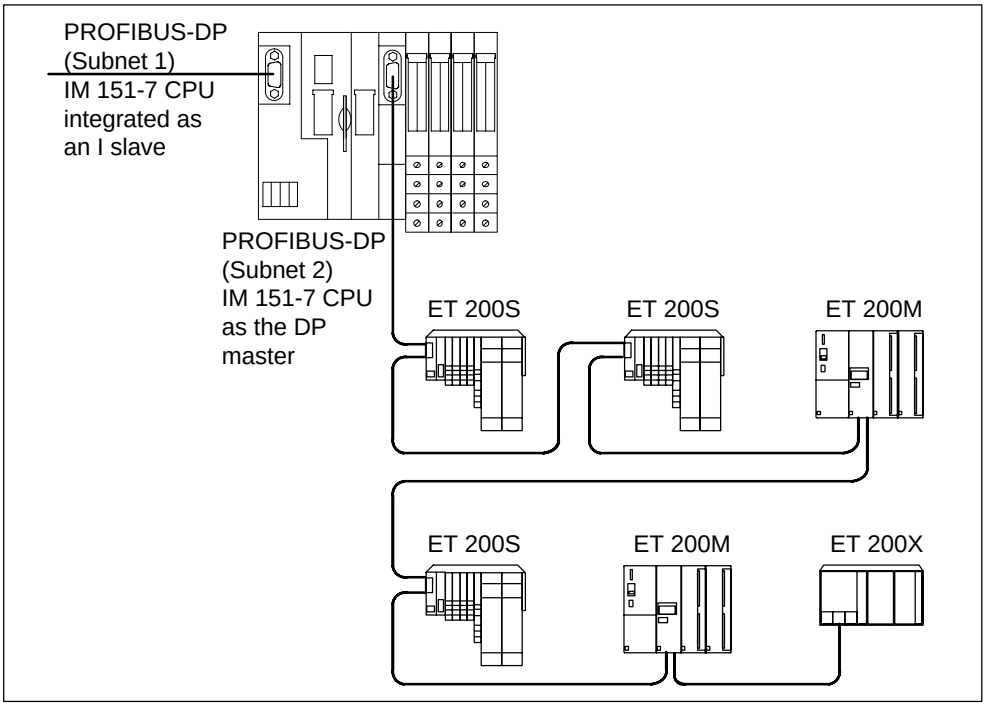


Figure 6-1 Example of the structure with the IM 151-7 CPU acting as the DP master

This chapter contains information on mounting the DP master module and on commissioning the IM 151-7 CPU as the DP master.

Chapter overview

In Section	Contents	Page
6.1	Mounting the DP master module	6-2
6.2	Starting up the IM 151-7 CPU as a DP master	6-3

6.1 Mounting the DP master module

When you expand your interface module IM 151-7 CPU with a DP master module, you can use the IM 151-7 CPU as a DP master.

Procedure

Step	Description
1	The IM 151-7 CPU is installed on the mounting rail.
2	Hang the DP master module onto the mounting rail to the right of the IM 151-7 CPU.
3	Swivel the DP master module down until it engages.
4	Slide the DP master module to the left until it engages audibly in the IM 151-7 CPU.
5	If required, mount terminal modules for power/electronics modules and slide the corresponding modules into the TM.

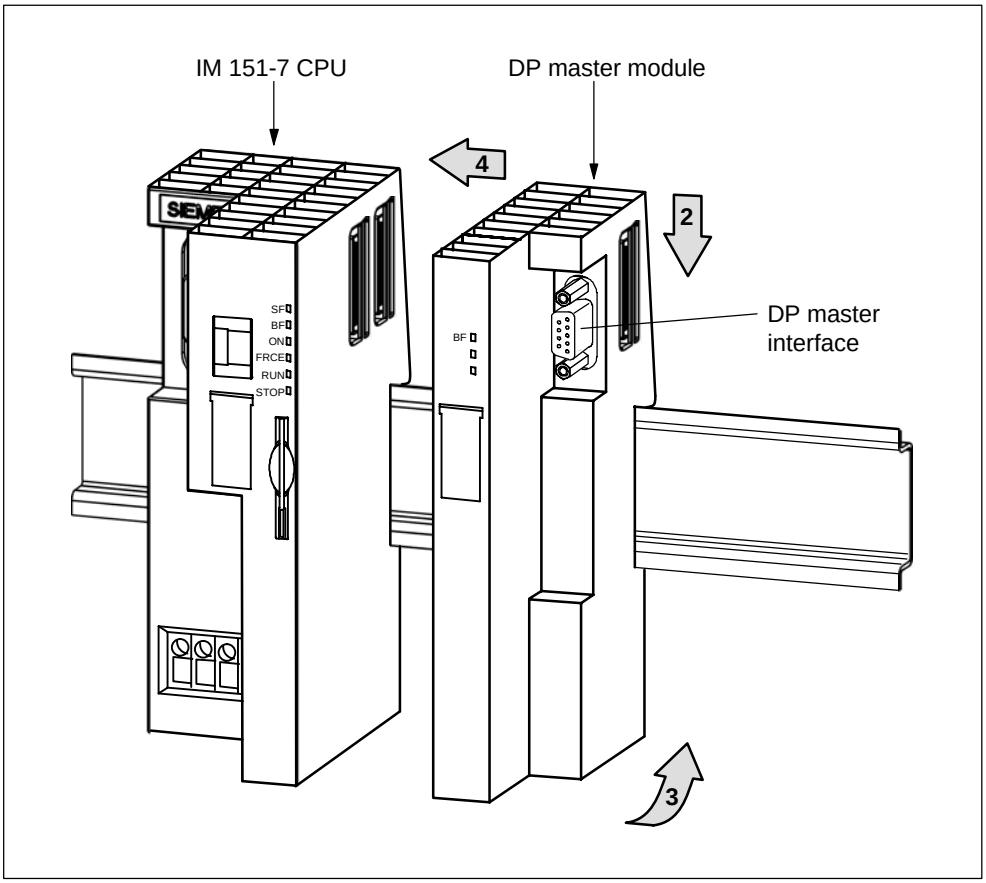


Figure 6-2 Mounting the DP master module

6.2 Commissioning the IM 151-7 CPU as a DP master

Prerequisites for commissioning

- The PROFIBUS subnet has been configured.
- The DP slaves have been prepared for operation (see the relevant DP slave manual).
- The IM 151-7 CPU must be configured as a DP master prior to commissioning. This means that you must use *STEP 7*
- to
  - configure the IM 151-7 CPU as a DP master (DP master interface),

Note

In the HW configuration, you must hang the DP master module in the station window separately as a submodule (X2).

  - assign the IM 151-7 CPU a PROFIBUS address,
  - assign the IM 151-7 CPU a master diagnosis address,
  - integrate DP slaves in the DP master system.

    Is an IM 151-7 CPU a DP slave?
    If so, you will find this DP slave in the PROFIBUS-DP catalog as a **pre-configured station**. In the DP master, you assign a slave diagnosis address to this DP slave CPU. You have to couple the DP master with the DP slave CPU and define the address ranges for the data exchange with the DP slave CPU.

Commissioning

Commission the IM 151-7 CPU as a DP master in the PROFIBUS subnet as follows:

1. The DP master module is mounted as described in Chap. 6.1.
2. Switch the supply voltage on.
3. Load the configuration of the PROFIBUS subnet (preset configuration) created with *STEP 7* into the IM 151-7 CPU with the PD.
4. Switch all DP slaves on.
5. Switch the IM 151-7 CPU from STOP to RUN.

Behavior of the IM 151-7 CPU upon commissioning

- The DP master module is mounted and the IM 151-7 CPU has been configured as a DP master
  ⇒ IM 151-7 CPU goes into RUN mode with master functionality
- The DP master module is mounted and the IM 151-7 CPU has not been configured as a DP master
  ⇒ IM 151-7 CPU goes into RUN mode without master functionality
- The DP master module is not mounted but IM 151-7 CPU has been configured as a DP master
  The behavior of the IM 151-7 CPU depends on the setting of the parameter **Start-up when preset configuration ≠ actual configuration**:
  - Start-up when the preset configuration ≠ actual configuration = yes
    ⇒ IM 151-7 CPU goes into RUN mode
    (see also "*Start-up of the IM 151-7 CPU as a DP master*")
  - Start-up when preset configuration ≠ actual configuration = no
    ⇒ IM 151-7 CPU remains in STOP mode
    (see also "*Start-up of the IM 151-7 CPU as a DP master*")

Initializing the DP master system

In addition, the initialization time monitor for the DP slave can be set with the **Monitoring time for transfer of parameters to modules** parameter.

This means that the DP slaves must be initialized and parameterized by the IM 151-7 CPU (as DP master) within the set time period.

Start-up of the IM 151-7 CPU as a DP master

When it starts up, the IM 151-7 CPU compares the preset configuration of the DP master system with the actual configuration.

If the preset configuration = the actual configuration, the CPU goes into RUN mode.

If the preset configuration ≠ the actual configuration, the behavior of the CPU depends on the setting of the **Start-up when the preset configuration ≠ actual configuration** parameter.

Start-up when preset configuration ≠ actual configuration = yes (default setting)	**Start-up when preset configuration ≠ actual configuration = no**
IM 151-7 CPU goes into RUN mode. (BF-LED on the DP master module flashes if all DP slaves cannot be addressed.)	IM 151-7 CPU remains in STOP mode and the BF-LED on the DP master module flashes after the set **Monitoring time for transfer of parameters to modules** has elapsed. A flashing BF-LED indicates that at least one DP slave cannot be addressed. Check whether all DP slaves are switched on and that they correspond to the defined configuration, or read out the diagnostic buffer using *STEP 7*.

Recognizing the operating modes of the DP slave (event recognition)

The table below shows how the IM 151-7 CPU identifies changes in operating mode and interruptions in data transfer as a DP master.

Table 6-1 Event recognition of the IM 151-7 CPU as a DP master

Event	What takes place in the DP master?
Bus interruption (short-circuit, connector removed)	• OB 86 is called and **Station failure** is reported (incoming event; diagnosis address of the DP slave assigned to the DP master) • In the case of I/O access: OB 122 is called (I/O access error)
DP slave: RUN → STOP	• OB 82 is called and **Module malfunction** reported (incoming event; diagnosis address of the DP slave assigned to the DP master; variable OB82_MDL_STOP=1)
DP slave: STOP → RUN	• OB 82 is called and **Module ok** reported. (outgoing event; diagnosis address of the DP slave assigned to the DP master; variable OB82_MDL_STOP=0)

Tip:
Always program OB 82 and OB 86 when commissioning the CPU as a DP master. This will allow you to detect and evaluate the faults and interruptions during data transfer.

Status/controlling, programming via PROFIBUS-DP

You can program the CPU or execute the PD functions listed in Chap. 4.4 via the DP master interface.

Note

The use of status and control via the DP master interface extends the DP cycle.

Equidistance

As of *STEP 7* V5.2, with the IM 151-7 CPU and DP master module, you can parameterize bus cycles of the same length (equidistant) for PROFIBUS subnets. You will find a detailed description of the functions in the *Online Help for STEP 7*.

PROFIBUS address of the DP master

- The permitted PROFIBUS-DP addresses are 1 to 125.
- Each address can be allocated only once on the PROFIBUS-DP.

Commissioning and Diagnostics

7

Configuring the IM 151-7 CPU with *STEP 7*

This chapter describes how to configure an ET 200S for the IM 151-7 CPU with *STEP 7*.

Resetting the memory of the IM 151-7 CPU

In certain situations you must reset the memory of the IM 151-7 CPU. This chapter describes these circumstances and the procedure for resetting the memory of the CPU component.

Diagnostic options

The ET 200S distributed I/O system is designed to make handling and commissioning as simple as possible. If a fault or an error should occur in spite of this, you can analyze it using the LEDs, the slave diagnosis and the diagnostic options in *STEP 7*.

Interrupt evaluation

To help you evaluate the **interrupts** of the ET 200S, we will examine the difference between these and the interrupts of the S7/M7 DP master and other DP masters.

Chapter overview

7.1 Configuring the IM 151-7 CPU

Configure the interface module IM 151-7 CPU as the following:

- I slave
- I slave and DP master
- Stand-alone module (MPI)
- Stand-alone module (MPI) and DP master

The IM 151-7 CPU is presented to the user in *STEP 7* as an S7-300 module that is always created together with a rack in an S7-300 station. Similarly, the module can only be deleted with the rack!

Expansion racks cannot be configured in an S7-300 station that contains an IM 151-7 CPU. The IM 151-7 CPU is positioned at slot 2 and receives an MPI/DP submodule. The first plug-in modules can be configured as of slot 4.

The following configuration options are available:

Table 7-1 Configuration options

Configuration environment	Configuration tool	Configurable operating mode
SIMATIC S7	*STEP 7* (HWConfig) V5.1 or higher + SP4	• Stand-alone (MPI) • IM 151-7 CPU as S7 slave
	STEP 7 (HWConfig) V5.2 or higher + SP1	IM 151-7 CPU with the DP master module as the DP master
SIMATIC S5	*COM PROFIBUS*	Fully configured and programmed IM 151-7 CPU, integrated as a standard intelligent slave via GSD in *COM PROFIBUS*
Non-Siemens systems	Non-Siemens tool	Fully configured and programmed IM 151-7 CPU, integrated as a standard intelligent slave via GSD in a non-Siemens tool

Note

If you wish to operate the IM 151-7 CPU as a standard I slave via the GSD file, then you should not activate the commissioning/test mode checkbox in the DP interface properties when configuring this slave CPU in *STEP 7*.

Information on configuring the DP master functionality is provided in Chapter 6.2.

Prerequisite

You have opened STEP 7 and are in the SIMATIC Manager of STEP 7.

Configuring the IM 151-7 CPU as an I slave

Proceed as follows:

1. Configure the IM 151-7 CPU as an S7-300 station.
   - Create a new station of the type **S7-300** (menu command **Insert → Station**).
   - Change to the hardware configuration window for this station.
   - In the ”Hardware Catalog” window, select the PROFIBUS-DP/ET 200S/IM 151-7 CPU folder.
   - Drag and drop the ”IM 151-7 CPU” object in the empty station window.
   - Configure the ET 200S with the required I/O modules.
   - Save the station (i.e. the ET 200S).
2. Configure a DP master (e.g. CPU with integrated PROFIBUS-DP interface or CP 342-5 with PROFIBUS-DP interface as of 6GK7 342-5DA01-0XE0, version 2) in another station in the same project.
3. Drag the ET 200S (with the IM 151-7 CPU) from the ”Hardware Catalog” window (from the **Configured Stations**) container and drop it on the icon for the DP master system.
4. Double-click the intelligent DP slave icon, and select the ”Interconnecting” tab. Specify on this tab which station is to represent the intelligent DP slave.
5. Select the intelligent DP slave, and click the ”Interconnect” button.
6. Select the (slave) configuration tab, and assign the master and slave addresses.
7. Click ”OK” to accept the settings.
8. The two stations must then be reloaded to start master-slave communication.

Configuration in a non-Siemens system

Using the DDB file you can also integrate the IM 151-7 CPU in non-Siemens systems as a standard I slave. In this case the diagnostic frame consists of the following:

- Station status
- Master PROFIBUS address
- Manufacturer ID
- Module diagnostics
- Module status

7.2 Resetting the memory of the IM 151-7 CPU

When do you reset the memory of the IM 151-7 CPU?

The memory of the IM 151-7 CPU must be reset

- to erase retentive areas (memory markers, times, counters)
- if the IM 151-7 CPU requests a memory reset by flashing the STOP LED at 0.5 Hz

The following are possible reasons for the MRES request:

- The ET 200S is starting up for the first time.
- Inconsistent memory areas
- The memory module (MMC) has been replaced.

How do you reset the memory?

There are two ways of resetting the IM 151-7 CPU:

Table 7-2 Ways to reset the memory

Resetting the memory with the mode selector	Resetting the memory with the PD
Described in this chapter.	Only possible during CPU STOP (see the PD manuals and the *STEP 7 Online Help*)

Resetting the memory of the IM 151-7 CPU with the mode selector

To reset the memory of the IM 151-7 CPU using the mode selector, proceed as follows (see also Figure 7-1):

1. Switch the mode selector to the STOP position.
2. Depress the mode selector in the MRES position. Hold the mode selector at this position until the STOP LED lights up for the second time (3 seconds) and then let it return to the STOP position.
3. Within 3 seconds, you must press the mode selector back to the MRES position and hold it in this position until the STOP LED flashes rapidly (at 2 Hz). When the IM 151-7 CPU has completed the memory reset, the STOP LED stops flashing and remains on.

The IM 151-7 CPU has reset the memory.

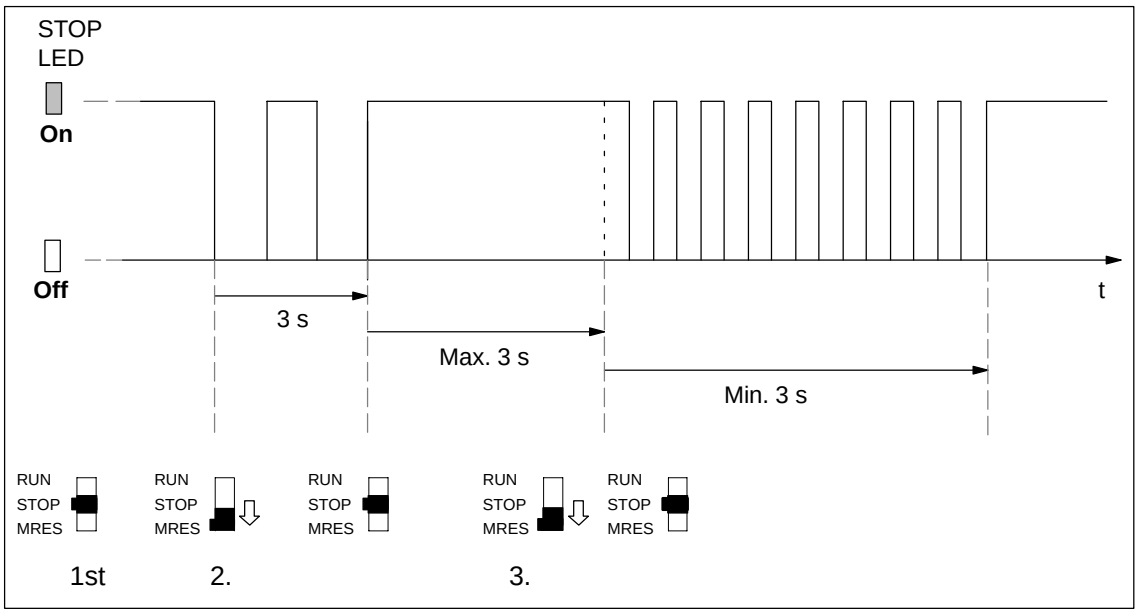


Figure 7-1 How to use the mode selector to reset the memory

Is the STOP LED not flashing at memory resetting?

Does the STOP LED not flash during memory reset or do other indicators come on? You must repeat steps 2 and 3. If the IM 151-7 CPU still doesn't execute a memory reset, you have to read out the diagnostic buffer of the CPU with the PD (see the *STEP 7* user manual).

What happens in the IM 151-7 CPU?

Table 7-3 Internal CPU events at memory resetting

Event	Response of the CPU in the IM 151-7 CPU
Sequence of operations in the IM 151-7 CPU	1. The CPU deletes the entire user program in the working memory and the RAM load memory. 2. The CPU deletes the retentive data. 3. The CPU tests its own hardware. 4. If you have inserted a memory module (micro memory card = MMC), the CPU copies the relevant contents of the module to the working memory.
Memory contents after reset	The CPU has the memory level "0". If a SIMATIC micro memory card is inserted, the user program is transferred back into the working memory.
What's left?	The contents of the diagnostic buffer and the runtime meter.

Note

If the CPU cannot copy the contents of the memory module (MMC) and requests a memory reset:

- Remove the MMC.
- Reset the CPU memory.
- Read out the diagnostic buffer.

You can read out the diagnostic buffer with the PD (see the *STEP 7 Online Help*).

7.3 Commissioning and startup of the IM 151-7 CPU as an I slave

An example of how to commission the IM 151-7 CPU as an I slave and its behavior when it starts up is described below.

Information on how to commission the IM 151-7 CPU as the DP master is found in Chapter 6.2.

Chapters 2.4, 5.2 and 8.5 contain information on stand-alone operation.

Prerequisites for commissioning

- The DP master (either an S7 DP master or another DP master) has been parameterized and configured.
- All other DP slaves have been parameterized and configured.

Commissioning the ET 200S

Commission the ET 200S distributed I/O system as follows:

1. Install the ET 200S distributed I/O system (see the *ET 200S Distributed I/O System* manual).
2. Wire the ET 200S distributed I/O system (see the *ET 200S Distributed I/O System* manual).
3. Assign a PROFIBUS address (see Chapter 4.3) and a slave diagnostic address (see Chapter 3.3) to the IM 151-7 CPU.
4. During configuration as an I slave, specify in the configuration software the address areas in the IM 151-7 CPU via which data exchange with the DP master is to take place (or use the ET 200S default setting; see Section 3.4).
5. Switch on the sensor supply voltage for the ET 200S.
6. If necessary, switch on the load voltage and the supply voltage for the motor starters.
7. If necessary, switch the IM 151-7 CPU to STOP mode.
8. Download the configuration for the IM 151-7 CPU to the ET 200S.
9. Switch the IM 151-7 CPU to RUN mode.

Starting behavior of the IM 151-7 CPU

On starting the IM 151-7 CPU following POWER ON, make sure that

- the terminating module is connected
- all terminal modules connected to the IM 151-7 CPU are fitted

Otherwise the IM 151-7 CPU will remain in the START mode.

Loading the user program

When commissioning the ET 200S, you can download the user program to the IM 151-7 CPU in the following ways:

- The program is downloaded from the PD/PC to the memory module (MMC) inserted in the IM 151-7 CPU by means of the ”Load User Program” function.

Note

This function does not delete retentive areas.

- The program is transferred from the PD/PC to the memory module (MMC). The memory module is then inserted in the IM 151-7 CPU and the memory reset request acknowledged.

See Section 8.3.

Tip: Programming OB 82 and 86 during commissioning

Always program OB 82 and OB 86 when commissioning as an I slave in the DP master and I slave using *STEP 7*. This will allow you to detect and evaluate the operating states of and interruptions during user data transfer (see Tables 7-6 and 7-7).

Note

Without configuration, a default start-up is possible if the power modules are switched on and all the modules are inserted.

Start-up

When the IM 151-7 CPU is switched to RUN mode, the following mutually independent operating mode transitions take place:

- The CPU switches from STOP to RUN mode.
- The IM 151-7 CPU starts user data transfer with the DP master on the PROFIBUS-DP.
- When the DP master module is plugged in, the IM 151-7 CPU starts user data transfer with the DP slaves on the PROFIBUS-DP.

7.4 Diagnostics using LEDs

LEDs

The RUN, STOP, ON, BF, SF and FRCE LEDs of the IM 151-7 CPU display important information on the states to the user.

The IM 151-7 CPU has the following 6 LEDs:

- ”SF” LED (**S**ystem **F**ault) for indicating the presence of a fault in the ET 200S
- ”BF” LED (**B**us **F**ault) for indicating faults on the PROFIBUS-DP

  On the IM 151-7 CPU: Bus fault on slave strand
  on the DP master module: Bus fault on master strand
- ”ON” LED for indicating that the ET 200S is connected to a power supply
- ”FRCE” LED for indicating that a force request is active.
- ”RUN” LED for indicating that the IM 151-7 CPU is in RUN mode
- ”STOP” LED for indicating that the IM 151-7 CPU is in STOP mode

The meaning of the LEDs for CPU functionality is described in detail in Section 8.2.

The ”BF” LED on the DP master module indicates in the DP master mode that errors have occurred in the PROFIBUS-DP.

”ON” LED is off

If the ”ON” LED is off, either no supply voltage or insufficient supply voltage is being applied to the electronic components/sensors of the ET 200S. The cause is likely to be a defective fuse or inadequate or nonexistent system voltage.

Diagnosis of DP functionality using the ”BF” and ”SF” LEDs

If the ”BF” and ”SF” LEDs light up or flash, the ET 200S is not configured correctly. The tables below shows you the possible error indications together with their meanings and the necessary action.

The table 7-4 shows the LED states for I slave operation. DP functionality is irrelevant in stand-alone operation (MPI), and a BF LED is not activated (there is no LED for transmission rate detection).

Table 7-4 LED display for PROFIBUS-DP (IM 151-7 CPU is an I slave)

”BF” LED On IM 151-7 CPU	”SF” LED On IM 151-7 CPU	Description	Cause	Error handling
On	On	No connection to the DP master	• IM 151-7 CPU is the active node ⇒ Bus short-circuit • IM 151-7 CPU is the passive node ⇒ Transmission rate detection No active node at bus, DP master does not exist or is switched off, or bus connection interrupted SF is on due to station failure	• Check that the connector for the PROFIBUS-DP is inserted correctly • Check whether the bus cable to the DP master is defective
Flashing	On	Parameter assignment error; there is no data exchange	• I slave not configured or incorrectly configured • Incorrect but permissible station address configured • Configured address areas of the actual configuration not identical to the target configuration • Station failure of a configured sender in direct data communication • DP master does not exist or is switched off	• Check the hardware of the ET 200S • Check the configuration and parameterization of the ET 200S • Check the setting for the configured address areas for the master
Off	On	Error in I slave: Diagnostics interrupt	Master in STOP	Switch the DP master to RUN mode.
Off	Off	Data exchange taking place	The target configuration and actual configuration of the ET 200S match.	

The table 7-5 shows the LED states for DP master operation.

Table 7-5 LED display for PROFIBUS-DP (IM 151-7 CPU is a master)

“BF” LED on DP master module	**“SF” LED on IM 151-7 CPU**	**Description**	**Cause**	**Error handling**
On	On	No connection to the DP slave	• Bus connection interrupted • Slave does not exist or is switched off • Bus short-circuit	• Check that the connector for the PROFIBUS-DP is inserted correctly • Check whether the bus cable to the DP master is defective • Evaluate the diagnosis. Reconfigure or correct the configuration.
Flashing	On	• There is no data exchange • Parameter assignment error	• A connected station has failed • At least one of the assigned slaves cannot be addressed • Configured address areas of the actual configuration not identical to the target configuration	• Check whether the bus cable is connected to the CPU or the bus is interrupted. • Wait until the CPU has run up. If the LED does not stop flashing, check the DP slaves or evaluate the diagnosis of the DP slaves. • Check the setting for the configured address areas for the master

7.5 Diagnostics via diagnostic address with *STEP 7*

Malfunctions that occur in the ET 200S are indicated by the ”SF” LED, and the cause is entered in the diagnostic buffer of the IM 151-7 CPU. Either the CPU enters STOP mode, or you can respond to errors by means of error or interrupt OBs in the user program.

To enable a response to be made, it must be possible to identify whatever caused the problem by means of a diagnostic address.

Diagnostic addresses

If you run the ET 200S with a DP master from the SIMATIC S7 range on the PROFIBUS-DP, diagnostic addresses are assigned in *STEP 7* as follows:

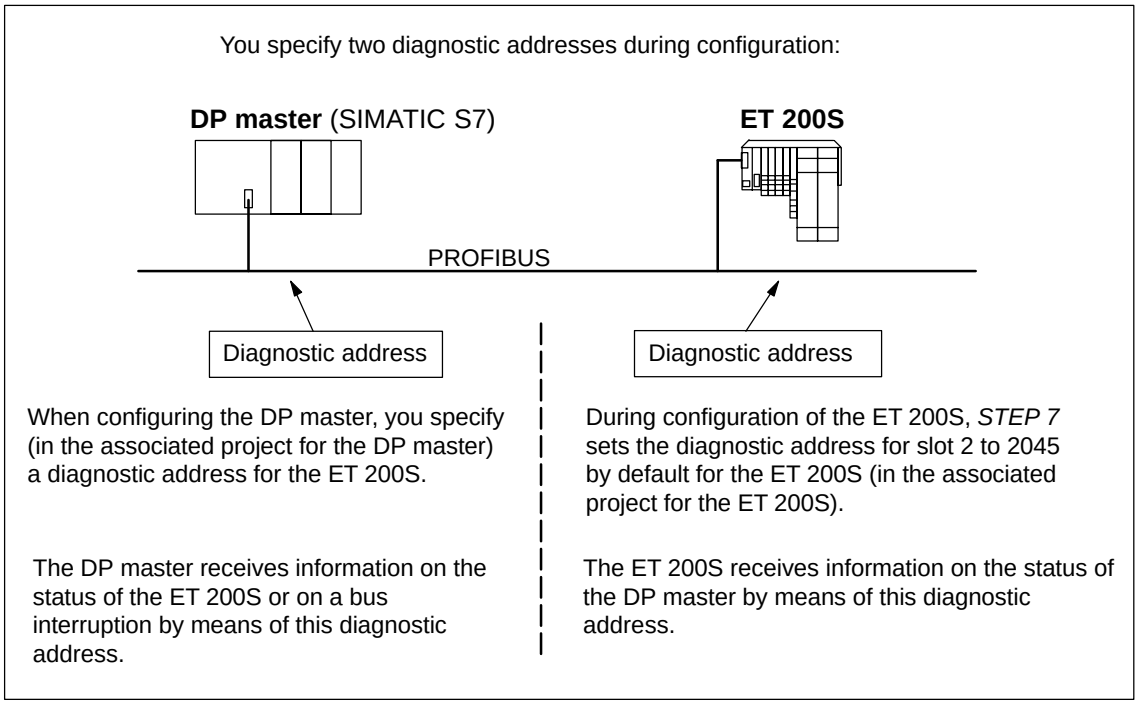


Figure 7-2 Diagnostic addresses for the DP master and ET 200S

Event identification

The table below indicates how the DP master or the IM 151-7 CPU of the ET 200S identifies changes in operating mode and interruptions in user data transfer.

Table 7-6 Responses to operating mode changes and interruptions in user data transfer in the DP master and the ET 200S with the IM 151-7 CPU as an I slave

Event	What happens ...	
	in the DP master	**in the IM 151-7 CPU**
Bus interruption (short-circuit, connector removed)	• OB 86 is called with the message *Station failure* (incoming event; diagnostic address of the IM 151-7 CPU) • With I/O access to transfer area: OB 122 is called (I/O access error)	• OB 86 is called with the message *Station failure* (incoming event; diagnostic address of the IM 151-7 CPU) • With I/O access to transfer area: OB 122 is called (I/O access error)
ET 200S: RUN → STOP	• OB 82 is called with the message *Faulty module* (incoming event; diagnostic address of the IM 151-7 CPU; variable OB82_MDL_STOP=1)	–
ET 200S: STOP → RUN	• OB 82 is called with the message *Module ok.* (outgoing event; diagnostic address of the IM 151-7 CPU; variable OB82_MDL_STOP=0)	–
DP master: RUN → STOP	–	• OB 82 is called with the message *Faulty module* (incoming event; diagnostic address of the IM 151-7 CPU; variable OB82_MDL_STOP=1)
DP master: STOP → RUN	–	• OB 82 is called with the message *Module ok.* (outgoing event; diagnostic address of the IM 151-7 CPU; variable OB82_MDL_STOP=0)

Evaluation in the user program

The table below indicates how you can evaluate e.g. RUN-STOP transitions in the DP master (CPU 315-2 DP; 6ES7 315-2AF03-0AB0) and in the ET 200S.

Table 7-7 Evaluation of RUN-STOP transitions in the DP master/ ET 200S with IM 151-7 CPU as the I slave

In the DP master		In the ET 200S (IM 151-7 CPU)
Diagnostic addresses: (example) Master diagnostic address=1023 Slave diagnostic address in the master system=**1022**		Diagnostic addresses: (example) Slave diagnostic address, slot 2=**2045** Master diagnostic address=not relevant
The CPU calls OB 82 with the following information: • OB82_MDL_ADDR:=**1022** • OB82_EV_CLASS:=B#16#39 (incoming event) • OB82_MDL_DEFECT:=Module fault Tip: This information is available in the diagnostic buffer of the CPU. In the user program, you should also program SFC 13 (”DPNRM_DG”) to read out the slave diagnosis.	←	CPU in the IM 151-7 CPU: RUN → STOP The CPU generates a diagnostic frame (slave diagnosis; see the *ET 200S Distributed I/O System* manual).
CPU: RUN → STOP	→	The IM 151-7 CPU calls OB 82 with information including the following: • OB82_MDL_ADDR:=**2045** • OB82_EV_CLASS:=B#16#39 (incoming event) • OB82_MDL_DEFECT:=Module fault Tip: This information is available in the diagnostic buffer of the CPU.

7.6 Slave diagnostics with IM 151-7 CPU used as an intelligent slave

Structure of the diagnostic frame

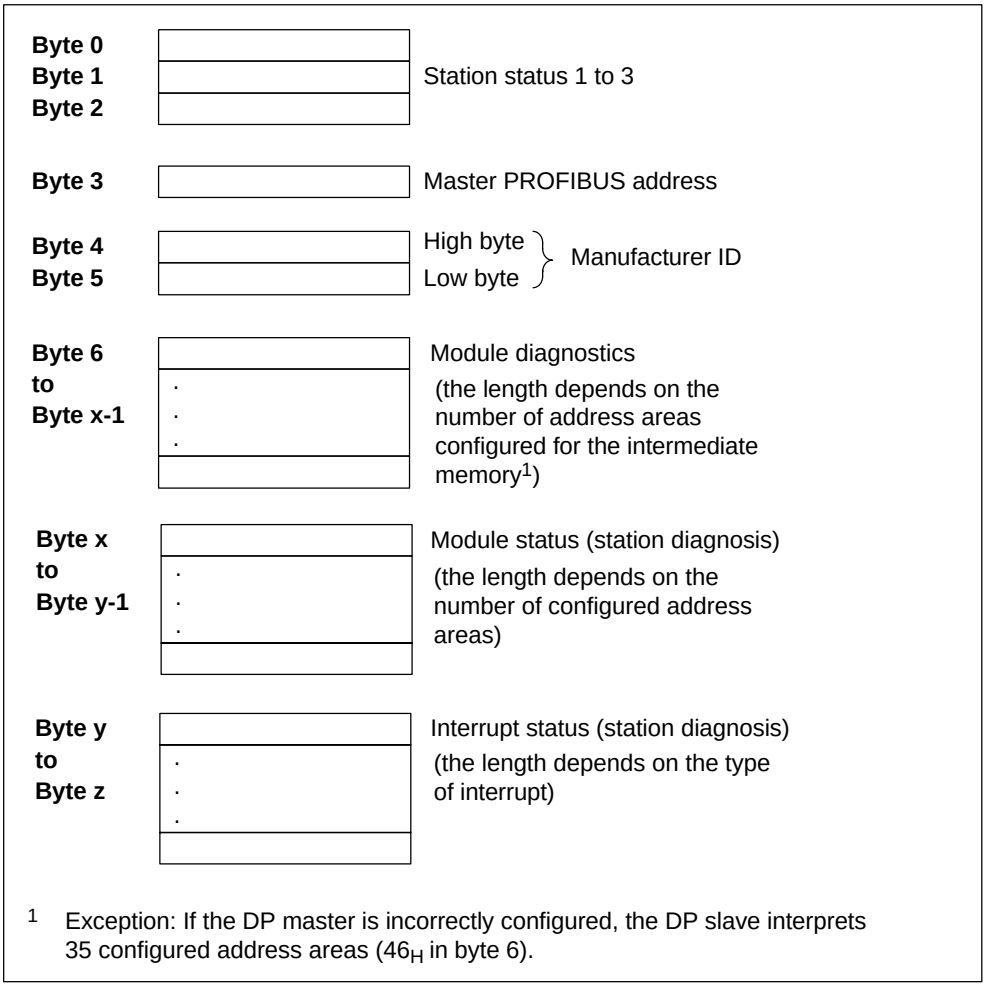


Figure 7-3 Format of the slave diagnostic data

7.6.1 Station status 1 to 3

Definition

Station status 1 to 3 provides an overview of the status of a DP slave.

Station status 1

Table 7-8 Structure of station status 1 (byte 0)

Bit	Description	Remedy
0	1: DP slave **cannot** be addressed by DP master.	• Is the correct DP address configured for the DP slave? • Is the bus connector inserted? • Does the DP slave have power? • Is the RS 485 repeater correctly set? • Execute a reset on the DP slave.
1	1: DP slave is not ready for data interchange.	• Wait; the DP slave is still doing its run-up.
2	1: The configuration data which the DP master sent to the DP slave do not correspond with the DP slave's actual configuration.	• Correct station type or correct configuration of the DP slave entered in the configuration software?
3	1: Diagnostic interrupt, generated by RUN-STOP transition of the CPU or by the SFB 75 0: Diagnostic interrupt, generated by STOP-RUN transition of the CPU or by the SFB 75	• You can read out the diagnostic data.
4	1: Function is not supported, for instance changing the DP address at the software level.	• Check the configuration data.
5	0: This bit is always "0".	–
6	1: DP slave type does not correspond to the software configuration.	• Was the configuration software set for the right station type? (parameter assignment error)
7	1: DP slave was parameterized by a different DP master to the one that currently has access to it.	• Bit is always "1" when, for instance, you are currently accessing the DP slave via the PD or a different DP master. The DP address of the master that parameterized the slave is located in the "Master PROFIBUS address" diagnostic byte.

Station status 2

Table 7-9 Structure of station status 2 (byte 1)

Bit	Description
0	1: DP slave must be parameterized again and reconfigured.
1	1: A diagnostic message has arrived. The DP slave cannot continue operation until the error has been rectified (static diagnostic message).
2	1: This bit is always ”1” when there is a DP slave with this DP address.
3	1: The watchdog monitor has been activated for this DP slave.
4	1: DP slave has received ”FREEZE” control command.
5	1: DP slave has received ”SYNC” control command.
6	0: The bit is always at 0.
7	1: DP slave is deactivated, that is to say, it has been removed from the scan cycle.

Station status 3

Table 7-10 Structure of station status 3 (byte 2)

Bit	Description
0 to 6	0: These bits are always ”0”.
7	1: • More diagnostic messages have arrived than the DP slave can buffer. • The DP master cannot enter all the diagnostic messages sent by the DP slave in its diagnostic buffer.

7.6.2 Master PROFIBUS address

Definition

The DP address of the DP master is stored in the master PROFIBUS address diagnostic byte:

- The master that parameterized the DP slave
- The master that has read and write access to the DP slave

Master PROFIBUS address

Table 7-11 Structure of the master PROFIBUS address (byte 3)

Bit	Description
0 to 7	DP address of the DP master that parameterized the DP slave and has read/write access to that DP slave.
	FF_H: DP slave has not been parameterized by a DP master.

7.6.3 Manufacturer ID

Definition

The manufacturer identification contains a code specifying the DP slave's type.

Manufacturer ID

Table 7-12 Structure of the manufacturer identification (bytes 4 and 5)

Byte 4	Byte 5	Manufacturer identification for
80_H	$E2_H$	IM 151-7 CPU

7.6.4 Module diagnostics

Definition

The module diagnosis indicates for which of the configured address areas of the intermediate memory an entry has been made.

Structure

The following figure shows the structure of the module diagnosis for the maximum number of configured address areas.

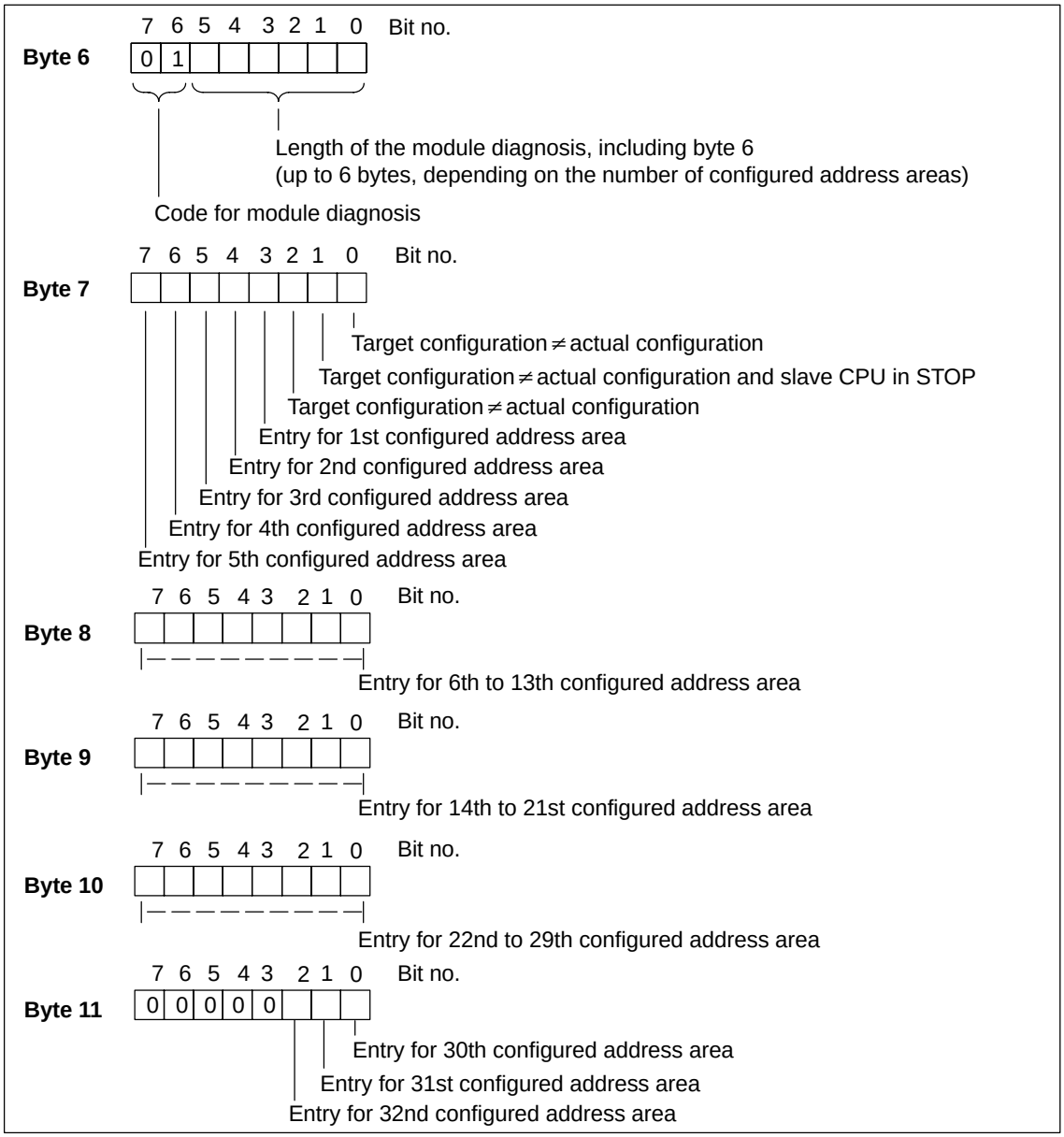


Figure 7-4 Structure of the module diagnosis for the IM 151-7 CPU

7.6.5 Module status

Definition

The module status indicates the status of the configured address areas and expands on the module diagnosis as regards the configuration. The module status begins after the module diagnosis and comprises max. 13 bytes.

Structure

The module status of the IM 151-7 CPU is structured as follows:

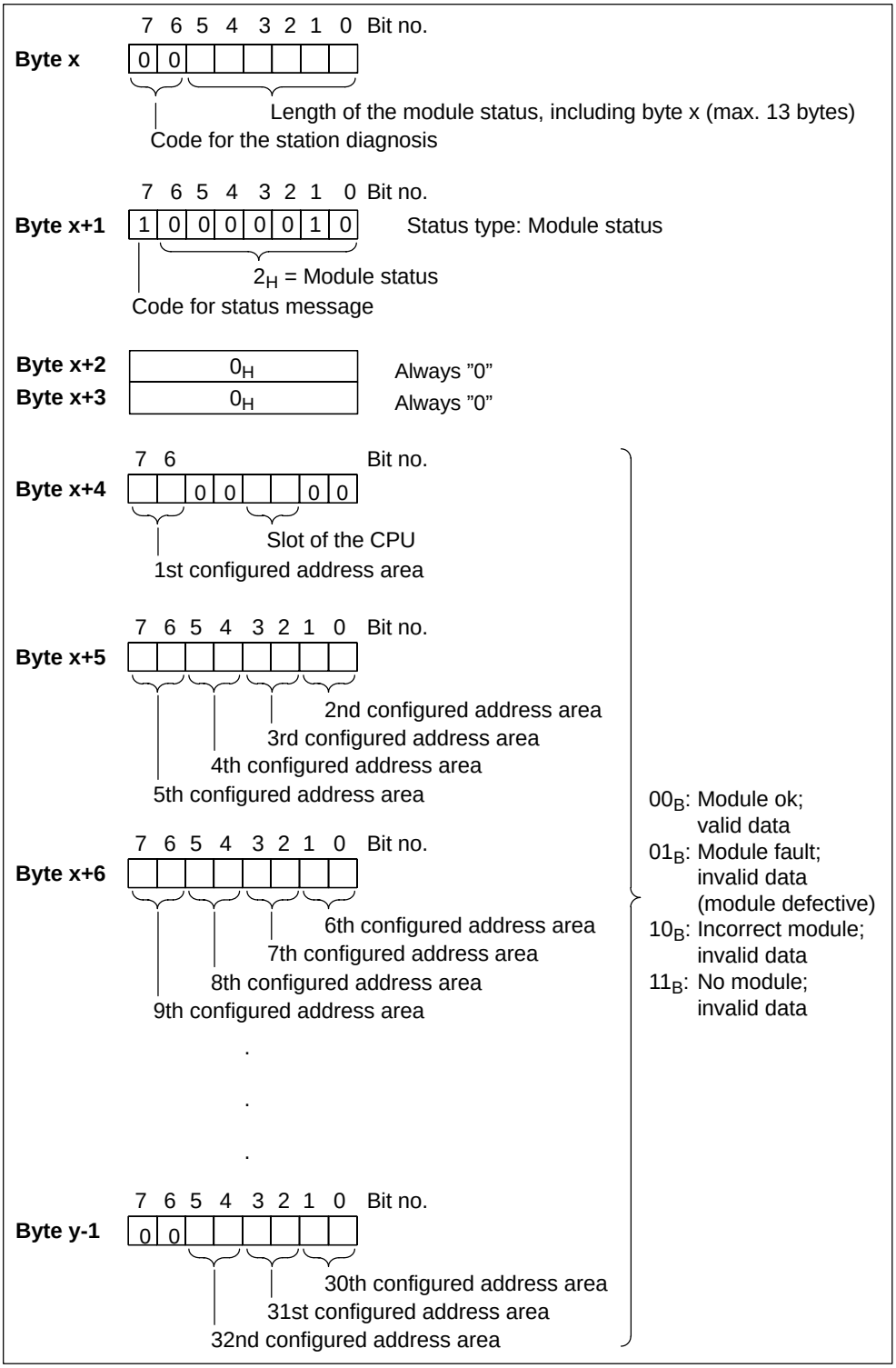


Figure 7-5 Structure of the module status

7.6.6 Interrupt status

Definition

The interrupt status of the station diagnosis provides detailed information about a DP slave. The station diagnosis begins at byte y and can comprise max. 20 bytes.

Structure

The following figure shows the structure and content of the bytes for a configured address area of the intermediate memory.

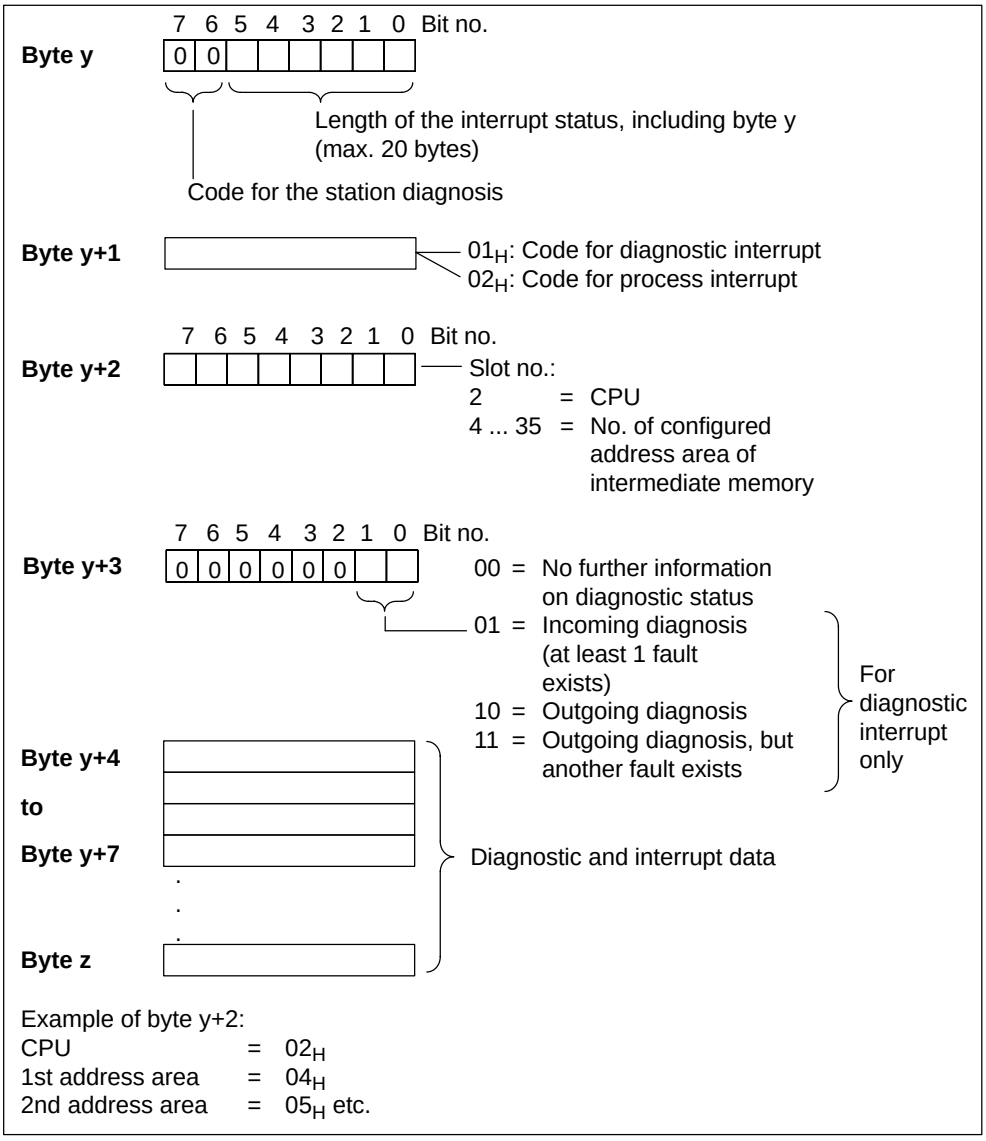


Figure 7-6 Structure of the interrupt status

Structure of the interrupt data with process interrupt (from Byte y+4 onwards)

With the process interrupt (in byte y+1, code 02_H stands for a process interrupt), the 4 byte interrupt information which you transfer in the intelligent slave with the SFC 7 ”DP_PRAL” and SFB 75 ”SALRM” when the process interrupt is generated for the master, is transferred from byte y+4 onwards.

Structure of the interrupt data when a diagnostic interrupt is generated by a mode change at the intelligent slave (from Byte y+4 onwards)

In the byte y+1, the code stands for diagnostic interrupt (01_H). The diagnostic data contains the 16 byte status information for the CPU. The following figure shows the assignment of the first 4 bytes of the diagnostic data. The next 12 bytes are always 0.

The content of these bytes corresponds to the content of data record 0 for diagnosis in *STEP 7* (in this case not all bits are assigned).

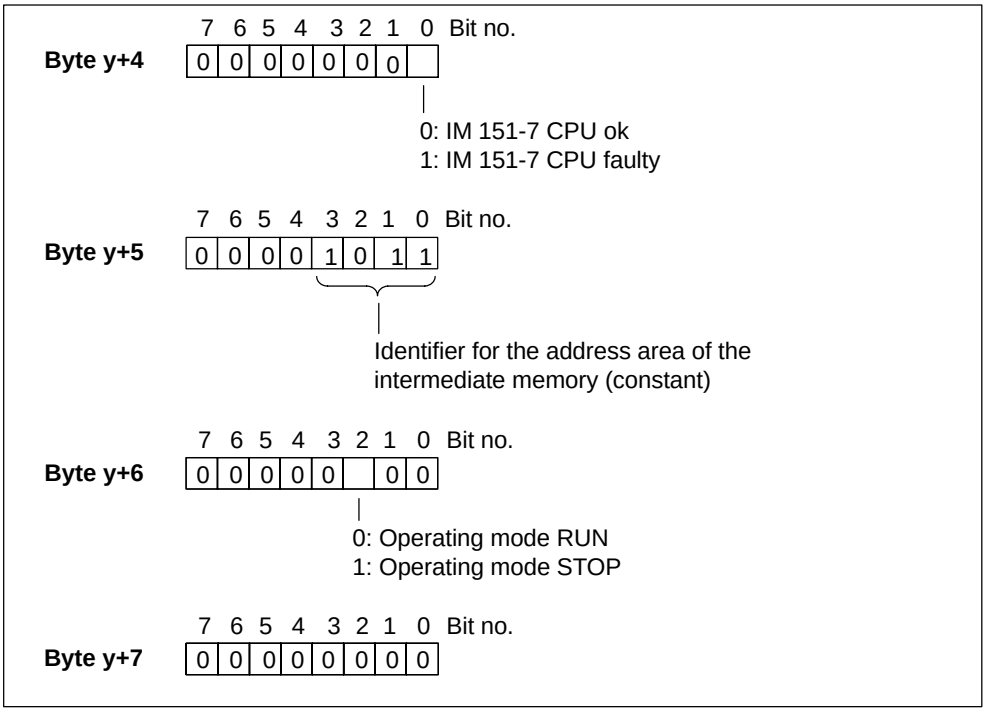


Figure 7-7 Byte y+4 to y+7 for the diagnostic interrupt (changed operating status of the intelligent slave)

Structure of the interrupt data when a diagnostic interrupt is generated by the SFB 75 in the intelligent slave (from Byte y+4 onwards)

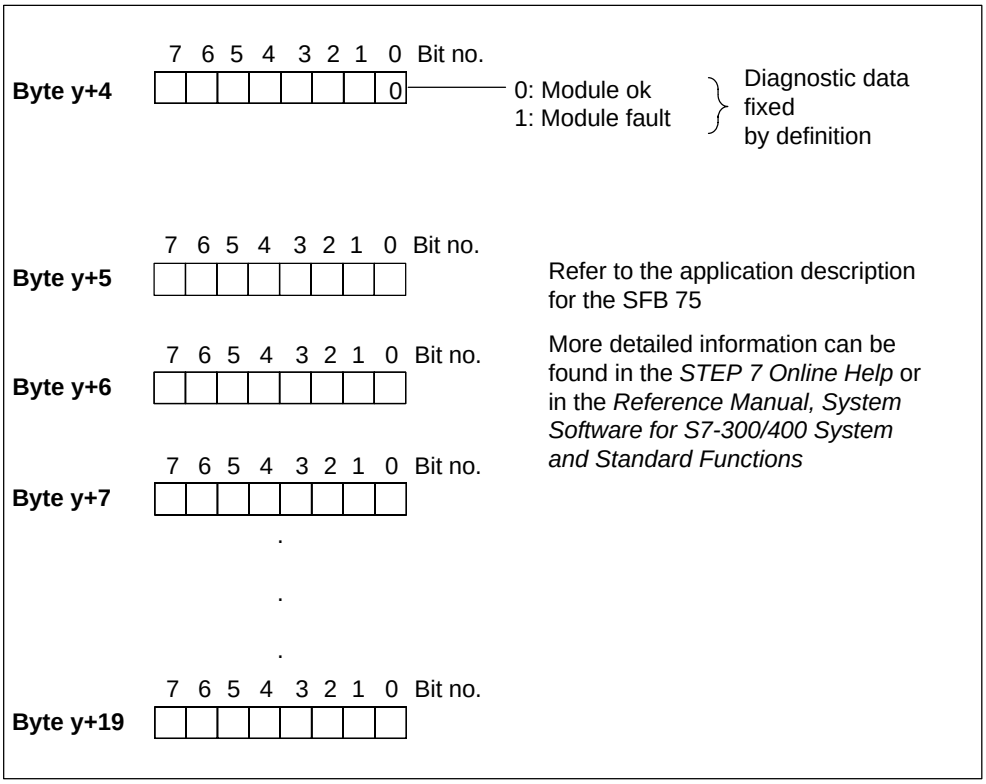


Figure 7-8 Byte y+4 to y+7 for diagnostic interrupt (SFB 75)

7.7 Diagnostic data of the electronic modules

7.7.1 Evaluating diagnostic data of the electronic modules in the user program

In this section

This section describes the structure of the diagnostic data in the system data. You must be familiar with this structure if you want to evaluate the diagnostic data of the electronic modules in the *STEP 7* user program.

The diagnostic data is located in the data records

The diagnostic data of a module can be up to 44 bytes in length and is located in data records 0 and 1:

- Data record 0 contains 4 bytes of diagnostic data describing the current status of a programmable logic controller.
  Data record 0 (DS0) is part of the header information of OB 82 (local data bytes 8 to 11).
- Data record 1 contains the 4 bytes of diagnostic data that is also contained in data record 0 **and**, in addition, up to 40 bytes of module-specific diagnostic data.

You can read out data record 0 and data record 1 using the SFC 59 ”RD_REC” and SFB 52 ”RDREC”.

More detailed information

A detailed description of the principles behind evaluating the diagnostic data of electronic modules in the user program and a description the SFCs which can be used for this purpose can be found in the *STEP 7* manuals.

Structure of the diagnostic data

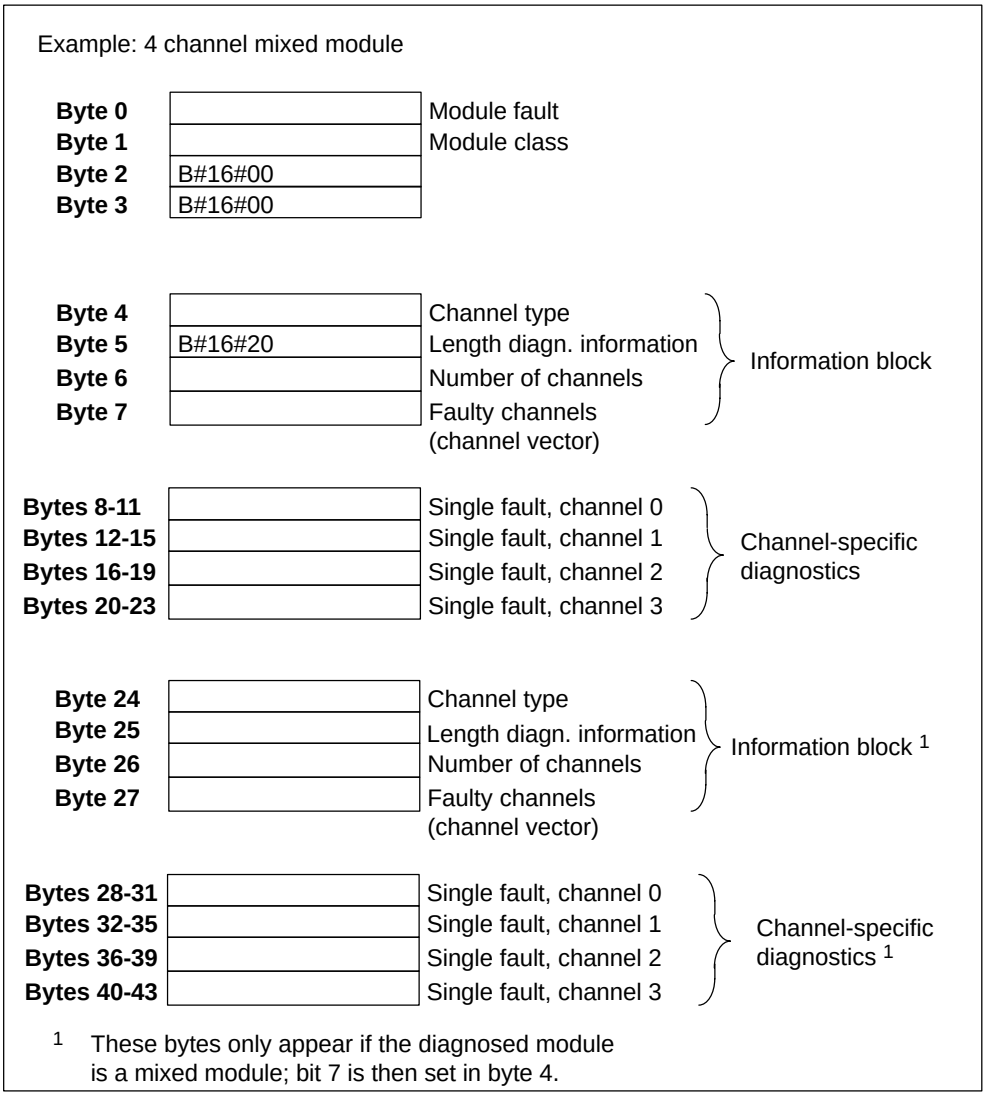


Figure 7-9 Structure of the diagnostic data using a 4 channel mixed module as an example

The number of channel-specific diagnostic bytes depends on the number of channels in the module. However, at least channel 0 must exist. The minimum length of data record 1 is therefore 12 bytes.

If, for example, you have a mixed module with 1 input channel and 2 output channels, the second information block begins at byte 12. In this example, the total length of the diagnostic data is 24 bytes.

7.7.2 Structure and content of the diagnostic data bytes 0 to 7

The structure and content of the individual bytes of the diagnostic data are described below. The following generally applies: If an error occurs, the corresponding bit is set to ”1”.

Bytes 0 and 1

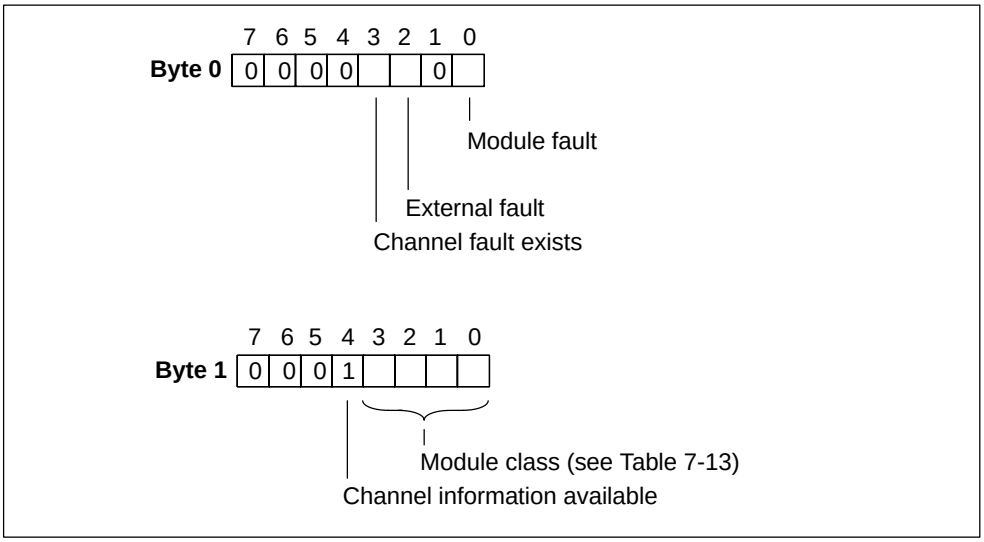


Figure 7-10 Bytes 0 and 1 of the diagnostic data

Module classes

The table below contains the identifiers for the module classes (bits 0 to 3 in byte 1).

Table 7-13 Identifiers of the module classes

Identifier	Module class
0101	Analog module
0110	CPU
1000	Function module
1001	Digital module (I/O with limited address area)
1100	CP
1101	PS

Bytes 2 and 3

These bytes are not used.

Bytes 4 to 7

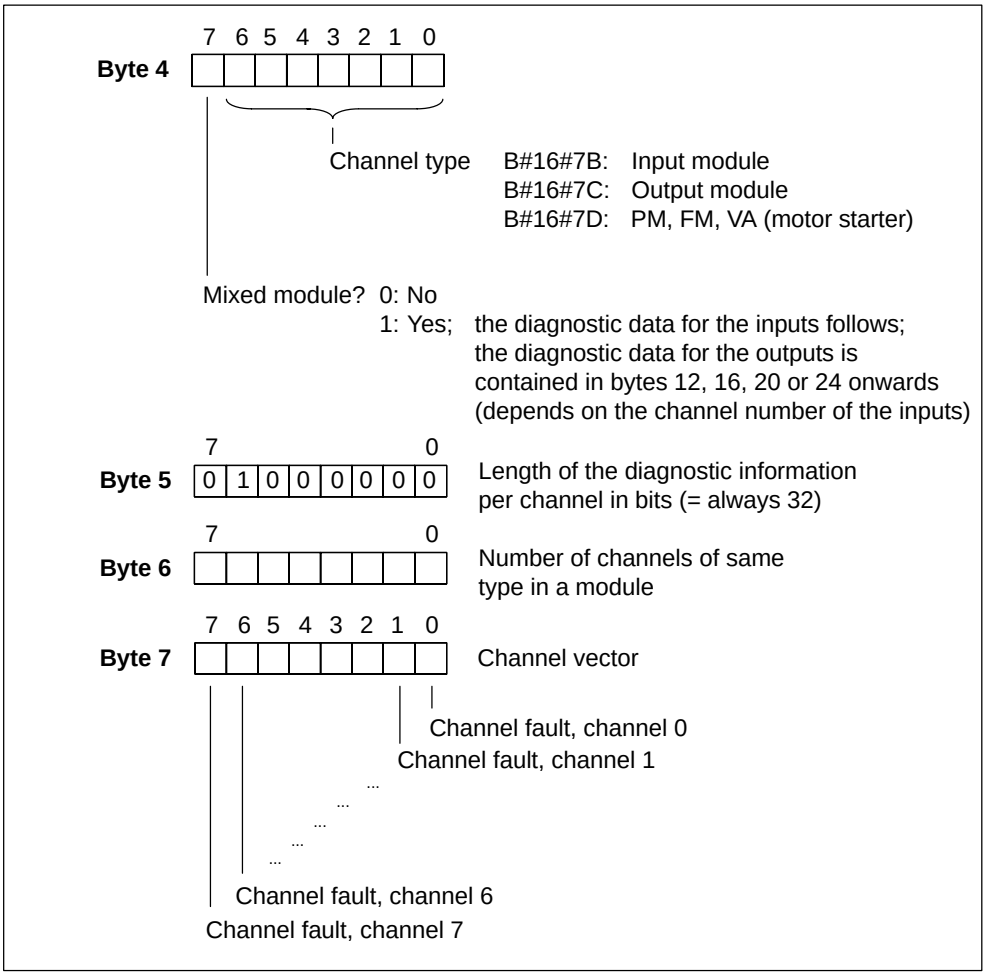


Figure 7-11 Bytes 4 to 7 of the diagnostic data

7.7.3 Channel-specific diagnostic data from byte 8 onwards

From byte 8 onwards, data record 1 contains the channel-specific diagnostic data. The figures below show the assignment of the diagnostic byte for a channel and a channel group of the specific module. The following generally applies: If an error occurs, the corresponding bit is set to ”1”.

Single fault of a channel

”Byte y” is the first of four bytes of the channel-specific diagnostics for a channel.

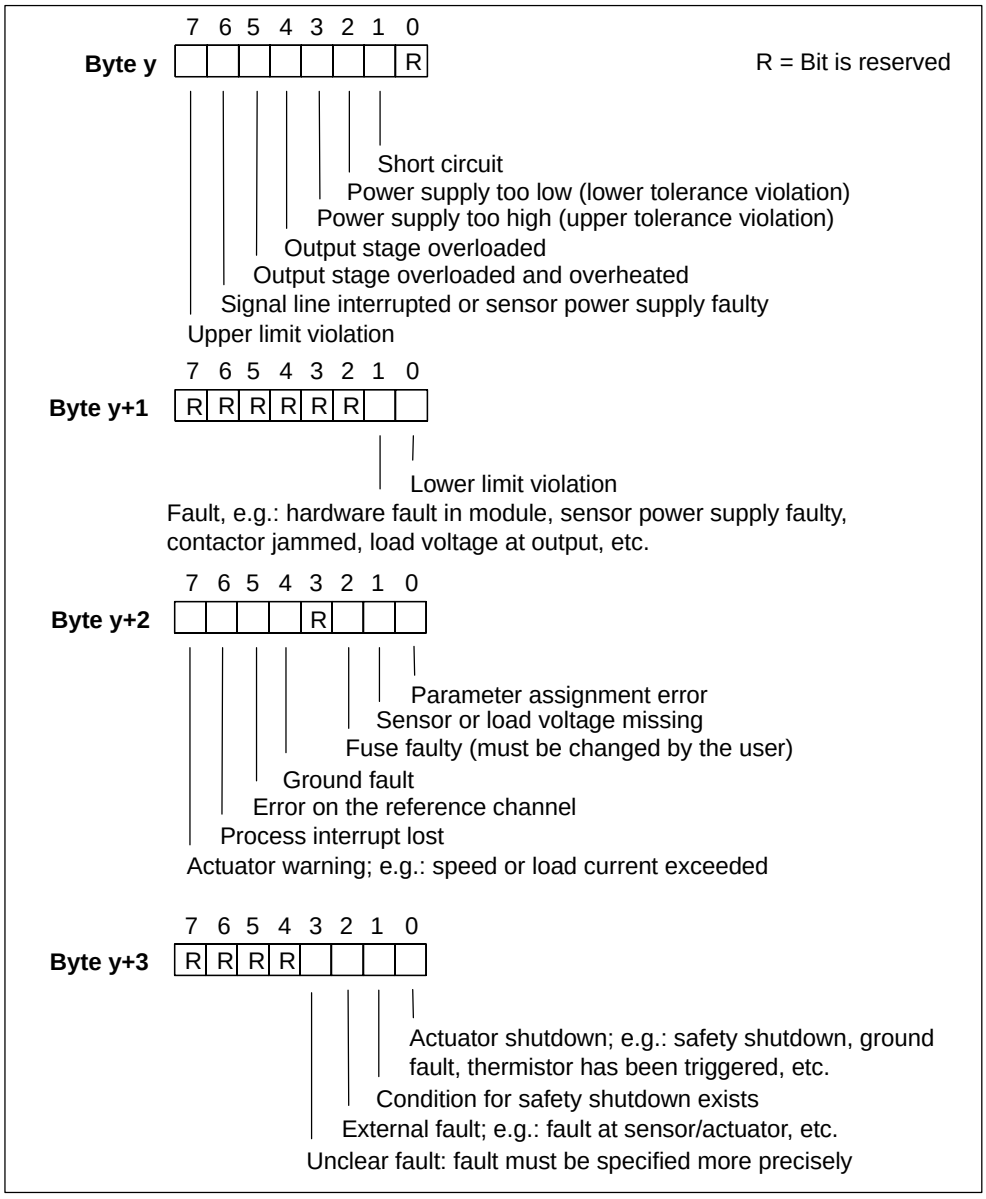


Figure 7-12 Single fault of a channel

7.7.4 Example: ET 200S module: 2 AI U (6ES7 134-4FB00-0AB0) each with diagnostics for channel 0 and 1

The table below shows examples of diagnostic message evaluation for the specified module.

Byte number	Value	Description
0	B#16#0D	Module fault, external fault, channel fault exists
1	B#16#15	Channel information exists; type class = analog module
2	B#16#00	Unassigned
3	B#16#00	Unassigned
4	B#16#7B	Input module, not a mixed module
5	B#16#20	= 32 bit diagnostic information per channel (constant)
6	B#16#02	The module has 2 channels
7	B#16#03	Channel fault at channel 0 and channel 1
8	B#16#80	Channel fault, channel 0: upper limit violation
9	B#16#00	Channel 0: no other fault
10	B#16#00	Channel 0: no other fault
11	B#16#00	Channel 0: no other fault
12	B#16#00	Channel 1: no fault
13	B#16#01	Channel fault, channel 1: lower limit violation
14	B#16#00	Channel 1: no other fault
15	B#16#00	Channel 1: no other fault

Functions of the IM 151-7 CPU

8

In this chapter

In this chapter you will find:

- Important features of the IM 151-7 CPU for the PROFIBUS-DP
- A list of the CPU functions of the IM 151-7 CPU that you can call with *STEP 7*, such as the integrated clock, blocks for the user program and parameters that can be set

Chapter overview

8.1 Data for the PROFIBUS-DP

Device master file

A DDB file contains all the slave-specific properties. The structure of the DDB file is defined in IEC 61784-1:2002 Ed1 CP 3/1.

You only need the DDB file if:

- You are using the ET 200S with a DP master from the SIMATIC S5 range (configuration with *COM PROFIBUS*)
- You are using the ET 200S with a non-SIMATIC DP master (configuration with a non-Siemens tool).

You can download the DDB file from the Internet. You will find all the DDB files under ”Downloads” on the SIMATIC Customer Support web site:

`http://www.ad.siemens.de/csi/gsd`

Important features

If you do not have the DDB file to hand, the following table lists the most important features of the IM 151-7 CPU.

Table 8-1 Attributes from the device database (DDB) file

Feature	DP code word to IEC 61784-1:2002 Ed1 CP 3/1	IM 151-7 CPU
Manufacturer ID	Ident_Number	$80E2_H$
Supports FMS	FMS_supp	No
Supports 9.6 kbaud	9.6_supp	Yes
Supports 19.2 kbaud	19.2_supp	Yes
Supports 45.45 kbaud	45.45_supp	Yes
Supports 93.75 kbaud	93.75_supp	Yes
Supports 187.5 kbaud	187.5_supp	Yes
Supports 500 kbaud	500_supp	Yes
Supports 1.5 Mbaud	1.5M_supp	Yes
Supports 3 Mbaud	3M_supp	Yes
Supports 6 Mbaud	6M_supp	Yes
Supports 12 Mbaud	12M_supp	Yes
Supports the FREEZE control command	Freeze_Mode_supp	Yes
Supports the SYNC control command	Sync_Mode_supp	Yes
Supports automatic transmission rate detection	Auto_Baud_supp	Yes
PROFIBUS address modifiable by software	Set_Slave_Add_supp	No
Length of user-specific parameter assignment data	User_Prm_Data_Len	3 bytes
User-specific parameter assignment data	User_Prm_Data	Yes

Table 8-1 Attributes from the device database (DDB) file, continued

Feature	DP code word to IEC 61784-1:2002 Ed1 CP 3/1	IM 151-7 CPU
Minimum interval between two slave list cycles	Min_Slave_Intervall	1(100µs)
Modular device	Modular_Station	1
Maximum number of modules	Max_Module	35
Maximum number of inputs in bytes	Max_Input_Len	244
Maximum number of outputs in bytes	Max_Output_Len	244
Maximum combined number of inputs and outputs in bytes	Max_Data_Len	488
Central display of vendor-specific status and error messages	Unit_Diag_Bit	Via ”ON” LED
Allocation of values in the station diagnostic field to texts	Unit_Diag_Area	Unassigned
Identifiers of all address areas for PROFIBUS	Module, End_Module	Yes
Allocation of vendor-specific error types in channel-specific diagnostic field to texts	Channel_Diag	No
Maximum length of the diagnostic data	Max_Diag_Data_Len	39 bytes

8.2 The mode selector and LEDs

Mode selector

The mode selector of the IM 151-7 CPU is designed as a 3-step toggle switch (see below):

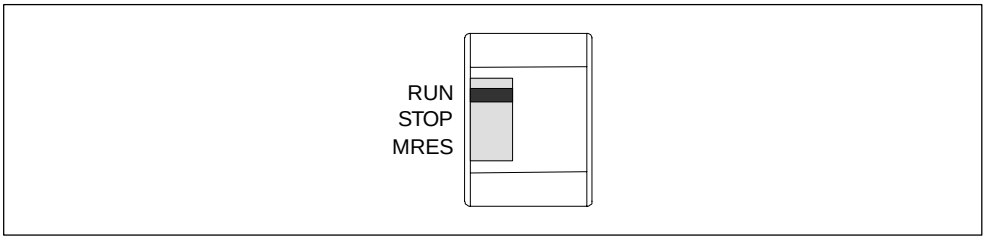


Figure 8-1 Mode selector

Positions of the mode selector

The positions of the mode selector are explained in the order in which they are arranged on the IM 151-7 CPU.

Table 8-2 Positions of the mode selector

Position	Description	Description
RUN	RUN mode	The CPU processes the user program.
STOP	STOP mode	The CPU processes no user program. Programs can: • Be read out from the CPU using a PD (CPU → PD) • Transferred to the CPU (PD → CPU)
MRES	Reset CPU memory	Momentary-contact position of the mode selector for resetting the CPU memory. You must adhere to a specific sequence when resetting the CPU memory using the mode selector (see Section 7.2)

Meanings of the LEDs for CPU functionality

For the IM 151-7 CPU there are 2 separate LEDs, which indicate the operating mode of the CPU:

- RUN
- STOP

You can obtain information on the power supply of the CPU, on force requests and on general errors via 3 additional LEDs.

Table 8-3 LEDs for CPU functionality

LED	Description	Description
ON (green)	Power on	**Comes on** when the supply voltage is applied to the CPU
RUN (green)	RUN mode	**Shines continuously** when the CPU is not processing the user program. **Flashes at 2 Hz** during CPU start-up: • For at least 3 secs, but the CPU start-up can also be shorter. • During the CPU start-up the STOP LED also lights up; when the STOP LED goes off, the outputs are enabled. **Flashes at 0.5 Hz** when the CPU has reached a breakpoint you have set. At the same time the STOP LED comes on.
STOP (yellow)	STOP mode	**Comes on** when the CPU • is not processing a user program. • Has reached a breakpoint you have set At the same time the RUN LED flashes at 0.5 Hz **Flashes** at 0.5Hz, when the CPU requests a memory reset (see Section 7.2).
FRCE (yellow)	Force request active	**Lights up** when a force request is active.
SF (red)	Group error	**Lights up** in the event of • Programming errors • Parameter assignment errors • Calculation errors • Timing errors • I/O errors • Hardware errors • Firmware errors To determine the exact nature of the error/fault, you have to use a PD and read out the contents of the diagnostic buffer.

Meanings of other LEDs

The LEDs ”SF” (from the PROFIBUS-DP viewpoint) and ”BF” are described in Section 7.4.

8.3 SIMATIC Micro Memory Card

Micro Memory Card

A SIMATIC Micro Memory Card (MMC) is used as a memory module for the IM 151-7 CPU. The MMC can be used as a load memory and portable data carrier. It is an essential requirement for operating the IM 151-7 CPU. The following data is stored on the MMC:

- User program (all blocks)
- Archives and recipes
- Configuration data (*STEP 7* projects)
- Data for an operating system update, operating system backup

Note

On **one** MMC you can store **either** user and configuration data or the operating system.

Copy protection

The MMC has an internal serial number for the purpose of providing MMC copy protection at the user level. You can read out this serial number from the SZL parts list, $011C_H$ Index 8, using the SFC 51 RDSYSST.

For example, you can program a STOP command in a know-how protected block for the event that the set and actual serial number of the MMC do not match.

More detailed information can be found in the *SZL parts list in the instructions list* or in the *System and Standard Functions* manual.

Features

The SIMATIC micro memory card ensures zero maintenance and retentivity for the IM 151-7 CPU. More detailed information can be found in Section 8.4.

Caution

The module content of a SIMATIC micro memory card can be corrupted if the card is removed while a write operation is being performed. The MMC must then be erased at the PD or formatted in the IM 151-7 CPU.

Never remove the MMC in RUN mode; it should only be removed when the IM 151-7 CPU is in the POWER OFF or STOP mode and only if the PD is not currently performing a write access operation. If in the STOP mode you are not sure whether or not the PD is currently performing a write access operation (e.g. loading/erasing a block), unplug the communication connections beforehand.

Service life of an MMC

The service life of an MMC mainly depends on the following factors:

1. The number of erasing and programming operations
2. External influences such as the ambient temperature

At an ambient temperature of up to 60° C, the service life of an MMC with max. 100,000 erase/write operations is 10 years.

Caution

To prevent data loss, never exceed the maximum number of erase/write operations.

Compatible SIMATIC Micro Memory Cards

The following memory modules are available:

Table 8-4 Available MMCs

Type	Order numbers
MMC 64k	6ES7 953-8LF00-0AA0
MMC 128k	6ES7 953-8LG00-0AA0
MMC 512k	6ES7 953-8LJ00-0AA0
MMC 2M	6ES7 953-8LL00-0AA0
MMC 4M	6ES7 953-8LM00-0AA0
MMC 8M	6ES7 953-8LP10-0AA0

The MMCs with a 4 MByte and 8 MByte memory are required for a firmware update.

Formatting the MMC prior to a memory reset

In certain exceptional cases, you have to format the MMC:

- The module type is not a user module.
- The MMC has not yet been formatted, is faulty or the data is corrupted.

The content of the MMC has been designated as invalid.

- The *Load User Program* operation has been interrupted by POWER OFF (see Special Measure).
- The *Promming* operation has been interrupted by POWER OFF (see Special Measure).
- A fault has occurred during evaluation of the module content prior to a memory reset.
- A fault has occurred during formatting, or formatting could not be performed.

If one of the above-described faults has occurred, the CPU also requests another memory reset after a memory reset has been performed. The content of the card is retained until the special measure has been completed, unless the *Load User Program* or *Promming* operations are interrupted by POWER OFF.

Description of the special measure:

When the IM 151-7 CPU requests a memory reset (the STOP LED flashes slowly), format it by operating the selector switch as follows:

1. Set the selector switch to the MRES position and hold it there (approx. 9 seconds) until the STOP LED remains lit (stops flashing).
2. Within the next 3 seconds you must release the selector switch and move it back to the MRES position. The STOP LED flashes during the formatting procedure.

Make sure that you perform the steps in the specified time, otherwise the MMC will not be formatted and will reassume the Memory Reset status.

The MMC is only formatted if a formatting condition (see above) exists and not e.g. when a memory reset is requested after a module is changed. In this case, switching to MRES only results in a standard memory reset whereby the content of the module remains valid.

Inserting/changing the card

The MMC is designed so that it can also be removed and inserted when the power is on. The IM 151-7 CPU must, however, be switched to the STOP mode (see the warning on page 8-6). The chamfered edge of the MMC prevents the card being inserted the wrong way round (reverse polarity protection).

There is an eject button on the memory card slot to enable you to remove the card easily. To eject the card, press the eject button with a small screwdriver or a ball-point pen.

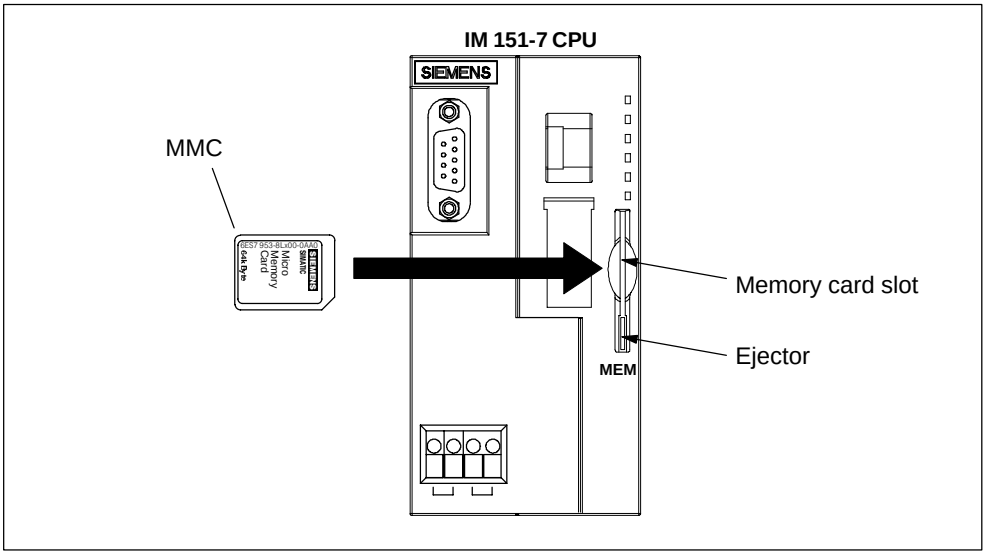


Figure 8-2 Position of the memory card slot for the MMC on the IM 151-7 CPU

If a new MMC is inserted in the memory card slot, the IM 151-7 CPU requests a memory reset.

Firmware update with MMC

To update the firmware, proceed as follows:

Table 8-5 Firmware update with MMC

Step	Action required	Action by the IM 151-7 CPU
1st	Transfer update files to a blank MMC using *STEP 7* and your programming device ($\geq$ 4 MB).	-
2.	Deenergize the IM 151-7 CPU and insert the MMC with the FW update	-
3.	Switch the power on	The IM 151-7 CPU automatically detects the MMC with the FW update and starts the FW update. All LEDs light up during the FW update. The STOP LED flashes after the FW update has been completed. In this way, the IM 151-7 CPU requests a memory reset.
4.	Deenergize the IM 151-7 CPU and remove the MMC with the FW update	-
5.	Switch on the power supply again.	The IM 151-7 CPU performs an automatic memory reset and is then ready for operation.

Backing up the operating system on the MMC

To back up the operating system, proceed as follows:

Table 8-6 Backing up the operating system

Step	Action required	Action by the IM 151-7 CPU
1st	Insert a new micro memory card (≥ 4 MB) in the CPU	The CPU requests a memory reset
2.	Hold the mode selector in the MRES position.	–
3.	Switch the power off then on, and keep the mode selector in the MRES position until...	... STOP, RUN and FRCE LEDs start flashing
4.	Move the mode selector to STOP	–
5.	Move the mode selector briefly to MRES, then let it snap back to STOP	The IM 151-7 CPU starts to back up the operating system on the MMC. All the LEDs light up during backup. The STOP LED flashes after the backup has been completed. In this way, the IM 151-7 CPU requests a memory reset.
6.	Remove the micro memory card	–

8.4 Memory concept

8.4.1 Memory areas of the IM 151-7 CPU

Organization

The memory of the IM 151-7 CPU can be divided into three areas:

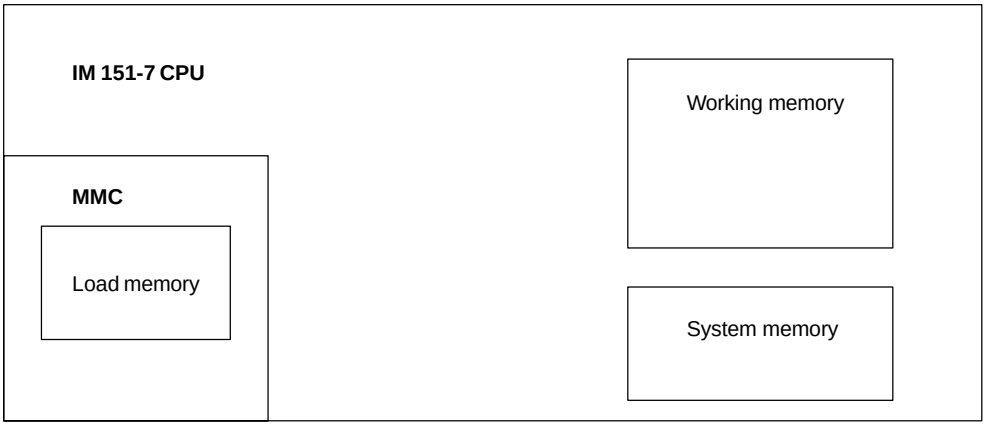


Figure 8-3 Memory areas of the IM 151-7 CPU

Load memory

The load memory is installed on a SIMATIC micro memory card (MMC). It is used to record code and data blocks as well as system data (configuration, connections, module parameters, etc.).

Blocks which are designated as non-process-related are recorded in the load memory.

The complete configuration data for a project can also be stored on the MMC.

Your program in the load memory (MMC) is always retentive. When downloaded, it is stored on the MMC such that it is unaffected by power failures and is not erased by memory resets.

Note

The IM 151-7 CPU can only be operated **with the MMC inserted**.

Working memory

The working memory is integrated on the CPU and cannot be expanded. It is used to process the codes and data of the user program. Program processing is only performed at the working memory and system memory.

The working memory of the CPU is retentive if the MMC is inserted.

Your data in the working memory is saved on the MMC if the power supply is interrupted.

System memory

The system memory is integrated on the CPU and cannot be expanded.

It contains

- the address areas ”memory markers”, ”timers” and ”counters”
- the process images of the inputs and outputs
- the local data

For memory markers, timers and counters, you can configure (Properties of the CPU, Retentivity tab) which parts are to be retentive and which parts are to be initialized with ”0” when a complete restart (warm restart) is performed.

The diagnostic buffer, MPI address (and transmission rate) as well as the run-time meter are generally stored in the retentive memory area on the CPU. Retentivity of the MPI address and transmission rate ensures that your CPU is still able to communicate following a power failure, a memory reset or the loss of communication parameters (by removing the MMC or erasing the communication parameters).

Retentivity

The IM 151-7 CPU has a retentive memory. The retentivity is provided on the MMC and CPU.

The retentivity means that the content of the retentive memory is retained even following POWER OFF and a restart (warm restart).

Retentive behavior of the memory objects

The following table shows the retentive behavior of the memory objects during the individual operating mode transitions.

Table 8-7 Retentive behavior of the memory objects

Memory object	Operating mode transition		
	POWER OFF/ POWER ON	STOP → RUN	Memory re-set
User program/data (load memory)	X	X	X
Current values of the DBs	Can be set in the DB properties using *STEP 7* V5.2 + SP1 (see below)		–
Memory markers, timers and counters configured as retentive	X	X	–
Diagnostic buffers, run-time meters	X	X	X
MPI address, transmission rate	X	X	X

x = retentive; – = not retentive

Retentive behavior of a DB in IM 151-7 CPU

For IM 151-7 CPU, you can use *STEP 7* (as of V5.2 + SP1) or the SFC 82 “CREA_DBL” (Parameter ATTRIB → Bit NON_RETAIN) to set whether, for POWER OFF/ON or STOP → RUN, a DB is to

- retain the current values (retentive DB) or
- adopt the initial values from the load memory (non-retentive DB).

Table 8-8 Retentive behavior of the DBs in IM 151-7 CPU

In the event of a POWER OFF/ON or a CPU restart, the DB is to	
Receive the initial values (non-retentive DB)	**Retain the last current values (retentive DB)**
Background: In the event of a POWER OFF/ON or a CPU restart (STOP → RUN), the current values of the DB are not retentive. The DB receives the initial values from the load memory.	Background: In the event of a POWER OFF/ON or a CPU restart (STOP → RUN), the current values of the DB are retained.
Prerequisite in *STEP 7*: • The “Non-Retain” checkbox is activated in the block properties of the DB or • A non-retentive DB was generated with the SFC 82 “CREA_DBL” and the associated block attribute (ATTRIB → Bit NON_RETAIN).	Prerequisite in *STEP 7*: • The “Non-Retain” checkbox is not activated in the block properties of the DB or • A retentive DB was generated with the SFC 82.

8.4.2 Memory functions

Introduction

You use the memory functions to generate, modify or erase user programs and individual blocks. The memory functions also allow you to archive your own project data in order to ensure the retentivity of your data.

General: Downloading the user program using the PD/PC

The complete user program is downloaded **to the IM 151-7 CPU from the MMC using the PD/PC**. In certain situations, all of the blocks stored in the load memory may be erased when the user program is downloaded.

In the load memory, blocks occupy the space specified under ”Load Memory Requirement” in the ”General Block Properties”.

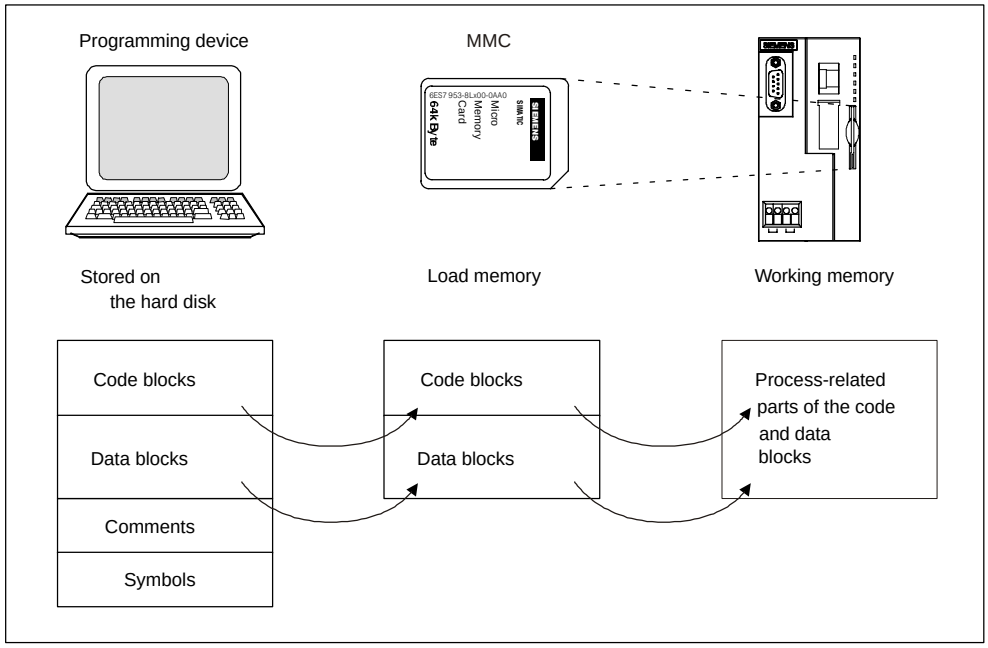


Figure 8-4 Load and working memory

The program cannot be started until all of the blocks have been loaded.

Note

The function is only permitted when the CPU is in STOP mode.

The load memory remains empty if loading could not be completed owing to a power failure or invalid blocks.

Downloading a user program to the MMC using the PD/PC

Case A: Downloading a new user program
You have created a new user program. Download the complete program to the MMC using the PD/PC.

Case B: Reloading blocks
You have already created a user program and downloaded it to the MMC (case A). You can then add further blocks to the user program. To do so, you do not have to download the complete user program to the MMC again; you only have to reload the new blocks to the MMC (this method shortens the loading time in the case of very complex programs!).

Case C: Overloading
In this case, you make changes to blocks of your user program. You then overload the user program or only the modified blocks to the MMC using the PD/PC.

Warning

When blocks/a user program are overloaded, all of the data stored on the MMC under the same name is lost.

When a block has been downloaded, the content in the case of process-related blocks is transferred to the working memory and activated.

Erasing blocks

When a block is erased, it is deleted from the load memory. It is possible to erase data blocks from the user program (SFC 23 ”DEL_DB”).

If memory in the working memory has been occupied by this block, this memory is released.

Uploading

Unlike downloading, uploading involves loading individual blocks or a complete user program **from the CPU to the PD/PC**. In this case, the blocks have the same content as when last downloaded to the MMC. Process-related data blocks are the exception; they receive the current values.

Uploading blocks or the user program from the CPU using *STEP 7* does not have any effect on the memory assignment of the CPU.

Compression

Compression fills any gaps between memory objects which are formed by loading and erasing operations in the load and working memory. Contiguous areas of free memory are then available.

Compression is possible when the CPU is in both the STOP and RUN mode.

Promming (RAM to ROM)

With promming, the current values of the data blocks are transferred from the working memory to the load memory where they serve as new initial values for the DB.

Note

The function is only permitted when the CPU is in STOP mode.

The load memory remains empty if the function could not be completed owing to a power failure.

Removing/inserting the MMC

The IM 151-7 CPU cannot run if there is no MMC inserted in the IM 151-7 CPU (no load memory available). Practical operation is only possible if an MMC has been inserted and a memory reset has been performed.

Removal and insertion of an MMC is detected by the IM 151-7 CPU in all operating modes.

Removing:

1. The IM 151-7 CPU must be in the STOP mode.
2. The PD must not be performing any write access operations (e.g. loading blocks)
3. When the MMC is removed, the IM 151-7 CPU requests a memory reset

Caution

The module content of a SIMATIC micro memory card can be corrupted if the card is removed while a write operation is being performed. The MMC must then be erased at the PD or formatted in the IM 151-7 CPU.

Never remove the MMC in RUN mode; it should only be removed when the IM 151-7 CPU is in the POWER OFF or STOP mode and only if the PD is not currently performing a write access operation. If in the STOP mode you are not sure whether or not the PD is currently performing a write access operation (e.g. loading/erasing a block), unplug the communication connections beforehand.

Inserting:

The MMC with the appropriate user program is inserted as follows:

1. Insert the MMC
2. The IM 151-7 CPU requests a memory reset
3. Acknowledge the memory reset

   If the IM 151-7 CPU requests a memory reset again owing to an incorrect MMC or an MMC with a firmware update being inserted, proceed as described in Section 8.3 under *Special Measure*.
4. Start the IM 151-7 CPU

Warning

Make sure that the MMC to be inserted contains the user program appropriate for the IM 151-7 CPU (for the system). An incorrect user program can have serious effects on processing.

Memory reset

A memory reset restores defined conditions following removal/insertion of the micro memory card so that the IM 151-7 CPU can be restarted (warm restart).

When the memory is reset, the memory administration system of the IM 151-7 CPU is reorganized. All blocks of the load memory are retained. All process-related blocks are transferred again from the load memory to the working memory; this initializes the data blocks in the working memory (they receive their initial values from the load memory again).

The memory reset procedure and the special points associated with it are described in Section 7.2.

Restart (warm restart)

- All DBs retain their current values.
- All retentive Ms, Cs and Ts retain their values.
- All non-retentive user data is initialized:
  - M, C, T, I, O with ”0”
- All processing levels start from the beginning.
- The process images are erased.

8.4.3 Address areas

Overview

The system memory of the IM 151-7 CPU id divided into address areas (see the table below). In your program, you use the appropriate operations to address the data directly in the respective address area.

Table 8-9 Address areas of the system memory

Address area	Description
Process image of the inputs	At the beginning of each OB 1 cycle, the IM 151-7 CPU reads the inputs out of the input modules and stores the values in the process image of the inputs.
Process image of the outputs	During the cycle, the program calculates the values for the outputs and stores them in the process image of the outputs. At the end of the OB 1 cycle, the IM 151-7 CPU writes the calculated output values to the output modules.
Memory markers	This area provides memory for intermediate results calculated in the program.
Timers	Timers are available in this area.
Counter	Counters are available in this area.
Local data	This memory area records the temporary data of a code block (OB, FB, FC) during the period in which this block is being processed.
Data blocks	See Section 8.4.4

The *Instruction list* tells you which address areas are possible with your CPU.

Process image of the inputs and outputs

If the address areas ”inputs” (I) and ”outputs” (O) are addressed in the user program, the signal states on the digital electronic modules are not checked, but instead a memory area in the system memory of the CPU is accessed. This memory area is referred to as the process image.

The process image is divided into two parts: the process image of the inputs and the process image of the outputs.

Advantages of the process image

The advantage of accessing the process image over accessing the electronic modules directly is that a consistent image of the process signals is available to the CPU for the duration of cyclic program processing. If a signal state at an input module changes during program processing, the signal state is retained in the process image until the process image is updated in the next cycle. Furthermore, accessing the process image requires much less time than accessing the electronic modules directly because the process image is located in the system memory of the CPU.

Updating the process image

The process image is updated by the operating system cyclically. The figure below shows the processing steps within a cycle.

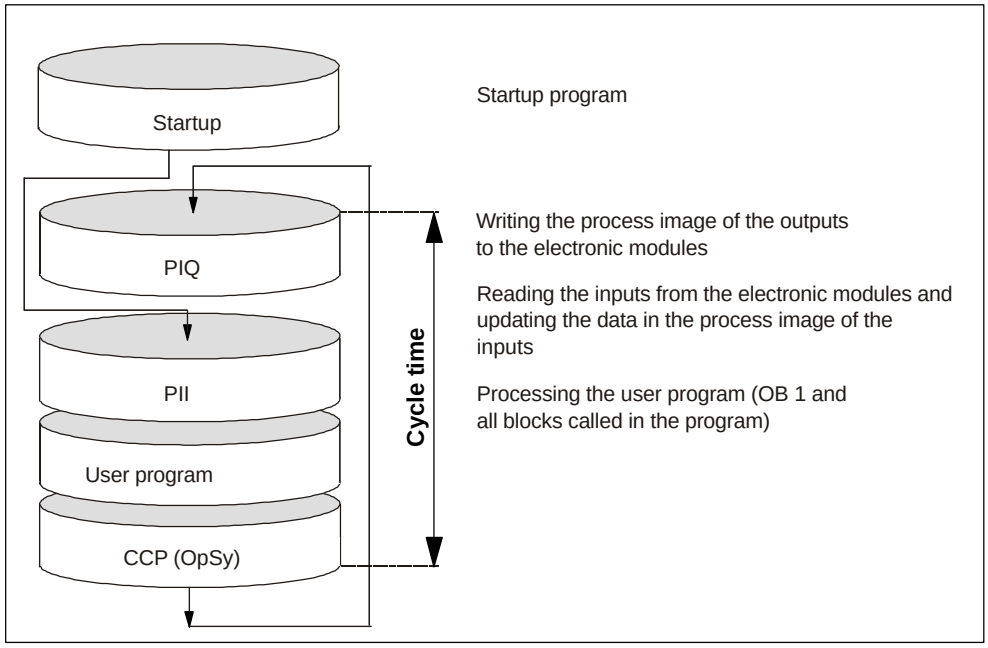


Figure 8-5 Processing steps within a cycle

Local data

The following are stored as local data:

- The temporary variables of code blocks
- The start information of the organization blocks
- Transfer parameters
- Intermediate results

Temporary variables

When creating blocks, you can declare temporary variables (TEMP) which are only available while the block is being processed and are then overwritten again. This local data has a fixed length for each OB. The local data must be initialized before the first read access operation. Furthermore, each organization block requires 20 bytes for its start information. Local data is accessed faster than data in the DBs.

The IM 151-7 CPU has memory for the temporary variables (local data) of blocks that have just been processed. This memory is divided equally between the priority classes. Each priority class has its own local data area.

Caution

All temporary variables (TEMP) of an OB and its subordinate blocks are stored in the local data. The local data area could overflow if you use a large number of nesting levels for block processing.
The IM 151-7 CPU changes to the STOP mode if you exceed the permitted quantity of local data in a priority class.
Take into consideration the amount of local data required by the synchronous fault OBs. In each case, the local data requirement is assigned to the responsible priority class.

8.4.4 Handling data in DBs

Recipes

A recipe is a collection of user data.

A simple recipe concept can be realized using non-process-related data blocks. The recipes should have the same structure (length). There should be one DB for each recipe.

Processing sequence of a recipe

The recipe is to be stored in the load memory:

- The individual data records of the recipes are created as non-process-related DBs using *STEP 7* and downloaded to the IM 151-7 CPU. The recipes therefore only occupy space in the load memory and not in the working memory.

Working with the recipe data:

- Calling the SFC 83 ”READ_DBL” from the user program causes the data record of the current recipe to be read out of the DB in the load memory and into a process-related DB in the working memory. As a result, the working memory only has to accommodate the data from one data record.

  The user program can now access the data of the current recipe.

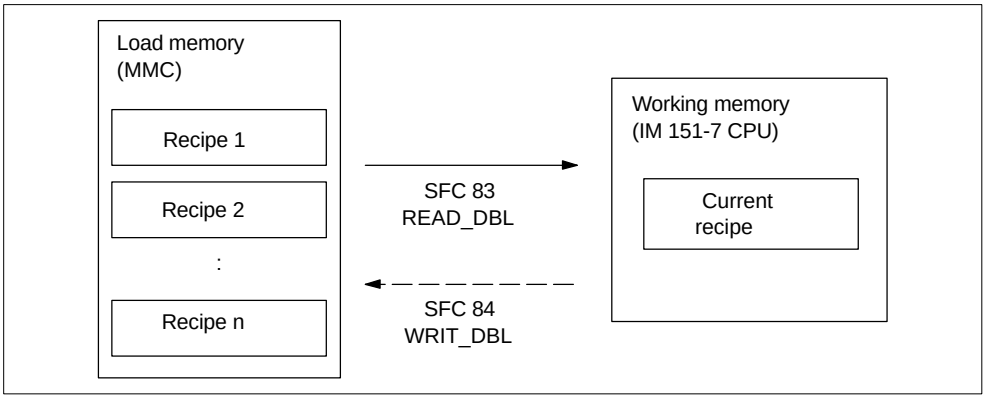


Figure 8-6 Handling recipe data

Writing back a modified recipe:

- Calling the SFC 84 ”WRIT_DBL” from the user program writes new or modified data records of a recipe, which are created during program processing, back to the load memory.

  The data written to the load memory is not erased by a memory reset and is transferrable.

  If modified data records (recipes) are to be stored on the PD/PC, they can be uploaded and stored there as a complete block.

Measured value archives

Measured values are created when the user program is processed by the IM 151-7 CPU. These measured values are to be archived and evaluated.

Processing sequence of a measured value archive

Collecting the measured values:

- The measured values in the working memory are collected in a DB (for alternating buffer operation in several DBs) by the IM 151-7 CPU.

Archiving the measured values:

- Calling the SFC 84 ”WRIT_DBL” from the user program moves the measured values in the DB to the load memory before the data volume can exceed the memory capacity of the working memory.

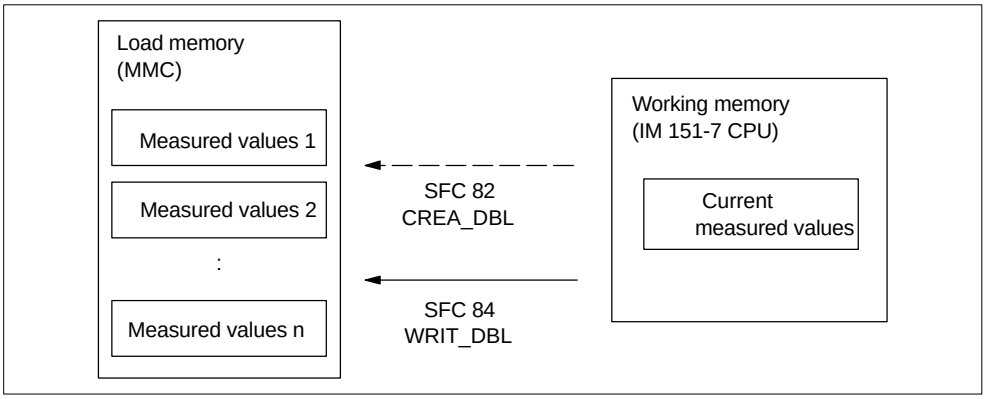


Figure 8-7 Handling measured value archives

- Calling the SFC 82 ”CREA_DBL” from the user program generates new (additional) DBs as non-process-related DBs in the load memory. These non-process-related DBs do not require space in the working memory.

Note

If a DB with the same name already exists in the load memory and/or working memory, the SFC 82 is terminated and an error display is generated.

The data written to the load memory is not erased by a memory reset and is transferrable.

Evaluating the measured values:

- The measured value data blocks stored in the load memory can be uploaded and evaluated by other communication partners (e.g. PD, PC, ...).

MMC access

Note

Active system functions SFC 82 to 84 (current access to the MMC) have a major influence on PD functions (e.g. status block, status variable, download block, upload block, open block). Their performance is typically 10 times lower (compared to non-active system functions).

To prevent data loss, never exceed the maximum number of erase/write operations of an MMC. Refer also to Section 8.3.

Caution

The module content of a SIMATIC micro memory card can be corrupted if the card is removed while a write operation is being performed. The MMC must then be erased at the PD or formatted in the CPU.

Never remove the MMC in RUN mode; it should only be removed when the CPU is in the POWER OFF or STOP mode and only if the PD is not currently performing a write access operation. If in the STOP mode you are not sure whether or not the PD is currently performing a write access operation (e.g. loading/erasing a block), unplug the communication connections beforehand.

8.4.5 Storing/download entire projects on/from Micro Memory Cards

You will find a detailed description of these functions in the *Online Help for STEP 7*.

8.5 Interfaces

The **IM 151-7 CPU** has a coexistent MPI/DP interface X1.
The **DP master module** has a PROFIBUS-DP master interface X2.

The interfaces are described below.

MPI interface

The MPI (Multi Point Interface) is the interface of the IM 151-7 CPU to a PD/OP and the interface allowing communication in an MPI network. The IM 151-7 CPU has an **MPI interface** which functions with RS 485.

The typical (default) transmission rate is 187.5 kbaud. The IM 151-7 CPU supports all MPI transmission rates.

The IM 151-7 CPU automatically sends its set bus parameters (e.g. the transmission rate) to the MPI interface. A programming device, for example, is thus supplied with the correct parameters and can automatically connect to an MPI subnetwork.

Note

You can only connect PDs to the MPI subnetwork during operation.
Other nodes (e.g. OP, TP, ...) should not be connected to the MPI subnetwork during operation, otherwise the transmitted data could be corrupted by interference noises or global data packets could be lost.

PROFIBUS-DP interface

The PROFIBUS-DP interface is mainly used to connect distributed I/Os. You use the PROFIBUS-DP to configure extended subnetworks. Transmission rates of up to 12 Mbaud are possible on the PROFIBUS.

The IM 151-7 CPU has a **PROFIBUS-DP interface**. This can be configured to be active or passive.

The IM 151-7 CPU as active station sends the set bus parameters (e.g. the transmission rate) to the PROFIBUS-DP interface. A programming device, for example, is thus supplied with the correct parameters and can automatically connect to a PROFIBUS subnetwork. The sending of bus parameters can be deactivated in the configuration settings.

DP master interface

The DP master interface on the DP master module is used to connect the distributed I/O (slaves). Transmission rates of up to 12 Mbaud are possible.

The DP master interface can be configured to be a DP master or to be inactive.

- As a DP master the interface requires a configuration. Slaves can be operated when the configuration is loaded; PD/OP functions and routing are possible.
- The interface is always inactive when there is not configuration.

Which devices can be connected to which interface?

Table 8-10 Connectable devices

MPI	PROFIBUS-DP	DP master interface
• PD/PC • OP/TP • S7-300/400 with MPI interface • S7-200 (only with 19.2 kbaud)	• DP master • Actuators/sensors • S7-300/400 with PROFIBUS-DP interface • PD/PC • OP/TP	• DP slaves • Actuators/sensors • S7-300/400 with PROFIBUS-DP interface • PD/PC • OP/TP

Further information

More detailed information on the individual connections can be found in the *Communication with SIMATIC* manual.

8.6 Clock

The IM 151-7 CPU has an integrated hardware clock.

Setting, reading and programming the clock

You set and read the clock using the programming device (see the *STEP 7*) User Manual) or program the clock in the user program using SFCs (see the *System and Standard Functions* Reference Manual and *Instruction list*).

Features

The table below indicates the features and functions of the clock.

When you parameterize the CPU in *STEP 7*, you can also set functions such as synchronization via the MPI interface and correction factors (refer to the *STEP 7* online help for information on how to do this).

Table 8-11 Features of the clock

Features	IM 151-7 CPU
Type	Hardware clock
Manufacturer setting	DT#1994-01-01-00:00:00
Backup	By means of integrated capacitor
Backup time	Typ. 6 weeks (at ambient temperature of 40 °C)
Run-time meter	1

Behavior of the clock when power is off

The clock of the CPU continues to run after the power has been switched off.

When the backup time has expired, the clock begins to run at the time at which the power was switched off.

8.7 S7 connections

Introduction

If S7 modules communicate with each other, a so-called S7 connection is built up between the modules. This forms the communication route.

Note

Global data communication does not require S7 connections.

Each communication connection requires S7 connection resources on the CPU; these must exist for the duration of the specific connection.

Therefore, a specific number of S7 connection resources is made available on each S7 CPU that are used by different communication utilities (PD/OP communication, S7 communication or S7 basic communication).

What are S7 connection points?

The S7 connection between communications-capable modules is established between connection points. Therefore, the S7 connection always has two connection points: an active and a passive connection point.

- The active connection point is assigned to the module that builds up the S7 connection.
- The passive connection point is assigned to the module that accepts the S7 connection.

Each communications-capable module can become a connection point of an S7 connection. At the connection point, the established communication connection always occupies **one** S7 connection of the specific module.

Pass-through point of an S7 connection

If you use the routing functionality, the S7 connection between two communications-capable modules is built up over several subnetworks. These subnetworks are connected to each other through a gateway. The module that is used to implement this gateway is called the router. Thus, the router is the pass-through point of an S7 connection.

The IM 151-7 CPU can be a router of an S7 connection if a DP master module has been plugged in. The IM 151-7 CPU can build up a maximum of 4 routing connections that do not restrict the quantity structure of the S7 connections.

Assigning S7 connections

The S7 connections on a communications-capable module can be assigned in various ways:

Reserving connections during configuration

- If, in *STEP 7*, a CPU is plugged in when the hardware is configured, S7 connections are automatically reserved on this CPU for both the PD and OP communication.
- For PD and OP communication and for S7 basic communication, the S7 connections can be reserved in *STEP 7*.

Assigning connections by programming

For the S7 basic communication, the connection is established via the user program. The CPU operating system initiates the build up of the connection and the corresponding S7 connections are assigned.

Assigning connections upon commissioning, testing and diagnosis

S7 connections for the PD communication are assigned by means of an online function on the Engineering Station (PD/PC with *STEP 7*):

- If an S7 connection was reserved in the CPU for PD communication when the hardware was configured, it is allocated to the Engineering Station, i.e. it is simply assigned.
- If all reserved S7 connections for PD communication have already been assigned and unreserved S7 connections are still fee, the operating system allocates a connection that is still free. If there are no longer any free connections, the Engineering Station cannot communicate online with the CPU.

Assigning connections for B&B utilities

S7 connections for the OP communication are assigned by means of an online function on the B&B Station (OP/TP/... with *ProTool*):

- If an S7 connection was reserved in the CPU for OP communication when the hardware was configured, it is allocated to the B&B Station, i.e. it is simply assigned.
- If all reserved S7 connections for OP communication have already been assigned and unreserved S7 connections are still fee, the operating system allocates a connection that is still free. If there are no longer any free connections, the B&B Station cannot communicate online with the CPU.

Sequence of the assignment of S7 connections

When the system is configured with *STEP 7*, parameter blocks are generated that are read out when the module is initialized. Thus, the module operating system reserves or assigns the corresponding S7 connections. This means, for example, that an Operator Station cannot access a reserved S7 connection for PD communication.

If the module (CPU) also has S7 connections that are not reserved, these are free to be used as required. The assignment of these S7 connections is in the order in which the requirements arises.

Example

If there is only one remaining free S7 connection on the CPU, you can connect a PD to the bus. The PD is then able to communicate with the CPU. However, the S7 connection is only assigned when the PD communicates with the CPU.

If you connect an OP to the bus at precisely the time when the PD is not communicating, the OP builds up a connection to the CPU. However, because an OP, unlike a PD, maintains a permanent communication connection, you will no longer be able to establish a connection via the PD.

Distribution of the S7 connections

The distribution of the S7 connections of the CPUs are shown in the table below:

Table 8-12 Distribution of the S7 connections

Communication utility	Distribution
PD communication OP communication S7 basic communication	So that the assignment of the S7 connections does not depend solely on the sequence in which the various communication utilities arise, S7 connections can be reserved for these utilities. For the PD and OP communication, at least one S7 connection is reserved for each as a preassignment. The table below and the technical data show the S7 connections that can be adjusted and the presettings for the IM 151-7 CPU. A “redistribution“ of the S7 connections is set up in *STEP 7* when the CPU is parameterized.
S7 communication Other communication connections (e.g. via CP 343-1 with data lengths > 240 byte)	The available S7 connections that have not been reserved for a specific utility (PD/OP communication, S7 basis communication) are assigned for this purpose.
Routing of PD functions (only for IM 151-7 CPU with DP master module)	The CPUs make available 4 connections for routing from PD functions. These connections are additional to the S7 connections.
Global data communication	This communication utility does **not** use S7 connections.

Availability of the S7 connections

The table below shows the S7 connections available on the IM 151-7 CPU.

Table 8-13 Availability of the S7 connections

Parameters	IM 151-7 CPU
Total number of S7 connections	12
• Reserved for PD communication	1 to 11 Default: 1
• Reserved for OP communication	1 to 11 Default: 1
• Reserved for S7 basic communication	0 to 10 Default: 0
• Free S7 connections	All unreserved S7 connections are shown as free connections.

Example of an IM 151-7 CPU

The IM 151-7 CPU makes 12 S7 connections available:

- Reserve 2 S7 connections for PD communication.
- Reserve 3 S7 connections for OP communication.
- Reserve 1 S7 connection for S7 basic communication.

There remain 6 S7 connections for other communication utilities, such as S7 communication, OP communication, etc.

More detailed information

- More detailed information on SFCs can be found in the *Instruction list*, in the *STEP 7 Online Help* or in the *System and Standard Functions* Reference Manual.
- More detailed information on communication can be found in the *Communication with SIMATIC* manual.
- More detailed information on routing can be found in Chapter 8.9 and in the *STEP 7 Online Help*.

8.8 Communication

Communication utilities of the IM 151-7 CPU

The selected communication utility influences

- the functionality available to the user
- whether or not an S7 connection is required
- the time at which the connection is set up

The user interface can be very different (SFC, SFB, ...) and also depends on the used hardware (SIMATIC-CPU, PC, ...).

The IM 151-7 CPU provides the following communication utilities:

Table 8-14 Communication utilities of the IM 151-7 CPU

Communication utility	Functionality	S7 connection	Via MPI	Via DP
PD communication	Commissioning, testing, diagnostics	Set up by the PD as soon as the utility is used	x	x
OP communication	Operating and monitoring	Set up by the OP when switched on	x	x
S7 basic communication	Data exchange	Programmed by means of blocks (parameters at SFC)	x	–
S7 communication	Data exchange	IM 151-7 CPU only as server; connection set up by the communication partner	x	x
Global data communication	Cyclic data exchange (e.g. memory markers)	Does not require an S7 connection	x	–
Routing of PD functions	For example, testing and diagnosis across network boundaries	Set up by the PD as soon as the utility is used	x	x

Chapters 4 and 1 contain information on network configuration and addressing.

PD communication

PD communication enables data exchange between engineering stations (e.g. PD, PC) and communications-capable SIMATIC modules. The utility is possible on MPI and PROFIBUS subnetworks. The transition between subnetworks is also supported.

PD communication provides functions which are required for downloading programs and configuration data, for executing tests and evaluating diagnostic information. These functions are integrated in the operating system of the SIMATIC S7 modules.

A CPU can maintain several online connections to one or more PDs simultaneously.

OP communication

OP communication enables data exchange between operator stations (e.g. OP, TP) and communications-capable SIMATIC modules. The utility is possible on MPI and PROFIBUS subnetworks.

OP communication provides functions which are required for operation and monitoring. These functions are integrated in the operating system of the SIMATIC S7 modules.

A CPU can maintain several connections to one or more OPs simultaneously.

S7 basic communication

S7 basic communication enables data exchange between S7-CPUs and communications-capable SIMATIC modules within an S7 station (acknowledged data exchange). Data is exchanged by means of non-configured S7 connections. The utility is possible on the MPI subnetwork or in the station of function modules (FM).

S7 basic communication provides functions which are required for data exchange. These functions are integrated in the operating system of the IM 151-7 CPU.

The user can use the utility via the ”System Function” user interface (SFC).

S7 communication

The IM 151-7 CPU can only be a server in S7 communication. The connection is always set up by the communication partner. The utility is possible on MPI and PROFIBUS subnetworks.

The utilities are processed by the operating system without an explicit user interface.

Global data communication

Global data communication enables the cyclic exchange of global data (e.g. I, O, M) between SIMATIC S7-CPUs (unacknowledged data exchange). The data is sent to all CPUs in the MPI subnetwork simultaneously by the CPU. The function is integrated in the operating system of the IM 151-7 CPU.

Transmit and receive conditions

You should observe the following conditions for communication in GD circles:

- The following must apply for the sender of a GD packet:
  $\text{Scan rate}_{\text{Sender}} \times \text{Cycle time}_{\text{Sender}} \geq 60\ \text{ms}$
- The following must apply for the recipient of a GD packet:
  $\text{Scan rate}_{\text{Recipient}} \times \text{Cycle time}_{\text{Recipient}} < \text{Scan rate}_{\text{Sender}} \times \text{Cycle time}_{\text{Sender}}$

A GD packet could be lost if you do not observe these conditions. Reasons for this are:

- the performance of the ”smallest” CPU in the GD circle
- global data is sent and received asynchronously by the sender and recipient

If in *STEP 7* you set: ”Send After Each CPU Cycle” and the CPU has a short CPU cycle (< 60 ms), the operating system could then overwrite a GD packet on the CPU which has not yet been sent. The loss of global data is indicated in the status field of the GD circle, provided that you have configured this option with *STEP 7*.

Scan rate

The scan rate indicates how many cycles the GD communication is divided into. You can set the scan rate when configuring the global data communication in *STEP 7*. If, for example, you select a scan rate of 7, global data communication only occurs every 7 cycles. This relieves the load on the CPU.

GD resources

The table below shows the GD resources of the IM 151-7 CPU.

Table 8-15 GD resources of the IM 151-7 CPU

Parameters	IM 151-7 CPU
Number of GD circles per CPU	Max. 4
Number of transmit GD packets per GD circle	Max. 1
Number of transmit GD packets for all GD circles	Max. 4
Number of receive GD packets per GD circle	Max. 1
Number of receive GD packets for all GD circles	Max. 4
Data length per GD packet	Max. 22 bytes
Consistency	Max. 22 bytes
Scan rate (default)	1 to 255 (8)

Routing

By means of an IM 151-7 CPU configured as a master and with *STEP 7* as of V5.2 + Service Pack 1, you can reach S7 stations across different subnetworks (MPI interface / PROFIBUS-DP interface) using a PD/PC.

For example, you can load user programs or a hardware configuration or perform testing and commissioning functions.

Note

If you use the your CPU as an I slave, the routing function can only be used when the DP interface is active.

Activate the Commissioning/Test Mode checkbox in the DP interface properties in *STEP 7* (see Section 4.1).

Detailed information can be found in the *Programming with STEP 7* manual or directly in the *STEP 7 Online Help*.

More detailed information

- More detailed information on SFCs can be found in the *Instruction list*, in the *STEP 7* Online Help or in the *System and Standard Functions* Reference Manual.
- More detailed information on communication can be found in the *Communication with SIMATIC* manual.

8.9 Routing

Access from a PD/PC to stations in another subnetwork

As of *STEP 7* V5.2 + Service Pack 1, you can reach S7 stations across subnetwork boundaries with the PD/PC when using IM 151-7 CPU and the DP master module, for example to load user programs or a hardware configuration or to perform testing and diagnosis functions.

The routing function lets you connect a PD anywhere in the network and establish a connection to any station that can be reached via the gateways.

The IM 151-7 CPU makes available 4 connections for routing PD functions. These connections are additional to the S7 connections.

Note

If you use the your IM 151-7 CPU as an I slave, the routing function can only be used when the DP interface is active.

Activate the Commissioning/Test Mode checkbox in the DP interface properties in *STEP 7*.

Detailed information can be found in the *Programming with STEP 7* manual or directly in the *STEP 7 Online Help*.

Gateway

The gateway from one subnetwork to one or more other subnetworks lies in a SIMATIC station that has interfaces to the respective subnetworks. Thus, in the diagram below, the CPU 31xC-2 DP acts as a router between subnetwork 1 and subnetwork 2.

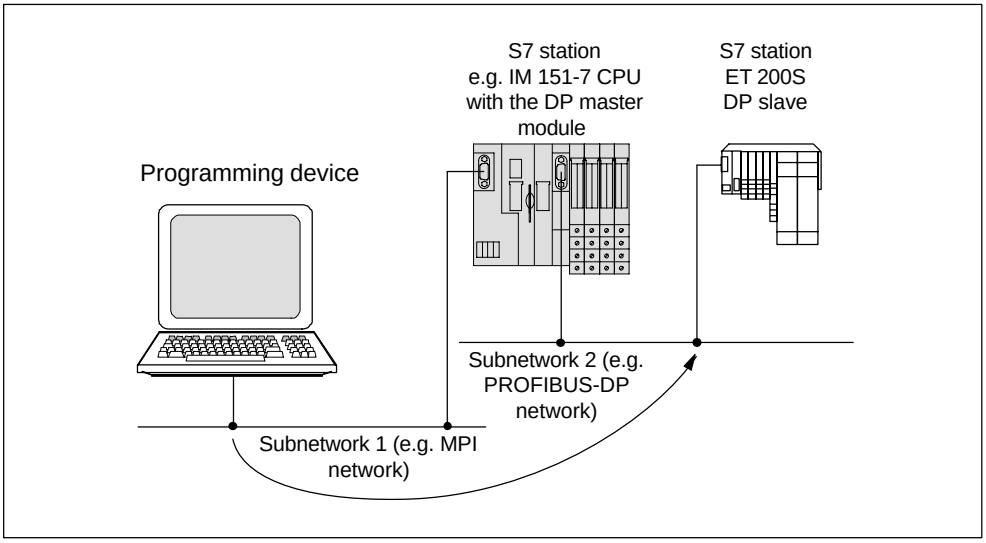


Figure 8-8 Routing gateway

Prerequisites

- The modules of the station are “capable of routing” (CPUs or CPs).
- The network configuration does not extend beyond the project boundaries.
- The modules have loaded in them the configuration information that contains the current “knowledge” on the entire network configuration of the project.

  Reason: All modules that participate in the gateway must contain information on which subnetworks can be reached along which routes (= routing information).
- The PD/PC with which you wish to establish a connection via a gateway must be assigned in the network configuration to the network to which it is actually physically connected.
- The CPU must be configured as a master.
- If the CPU is configured as a slave, the Commissioning/Test Mode functionality must be activated in the DP interface properties for the DP slave in *STEP 7*.

Application example: TeleService

As an application example, the following figure shows the teleservice of an S7 station by a PD. The connection is established across subnetwork boundaries and by means of a modem connection.

The lower section of the figure show how easily this can be configured in *STEP 7*.

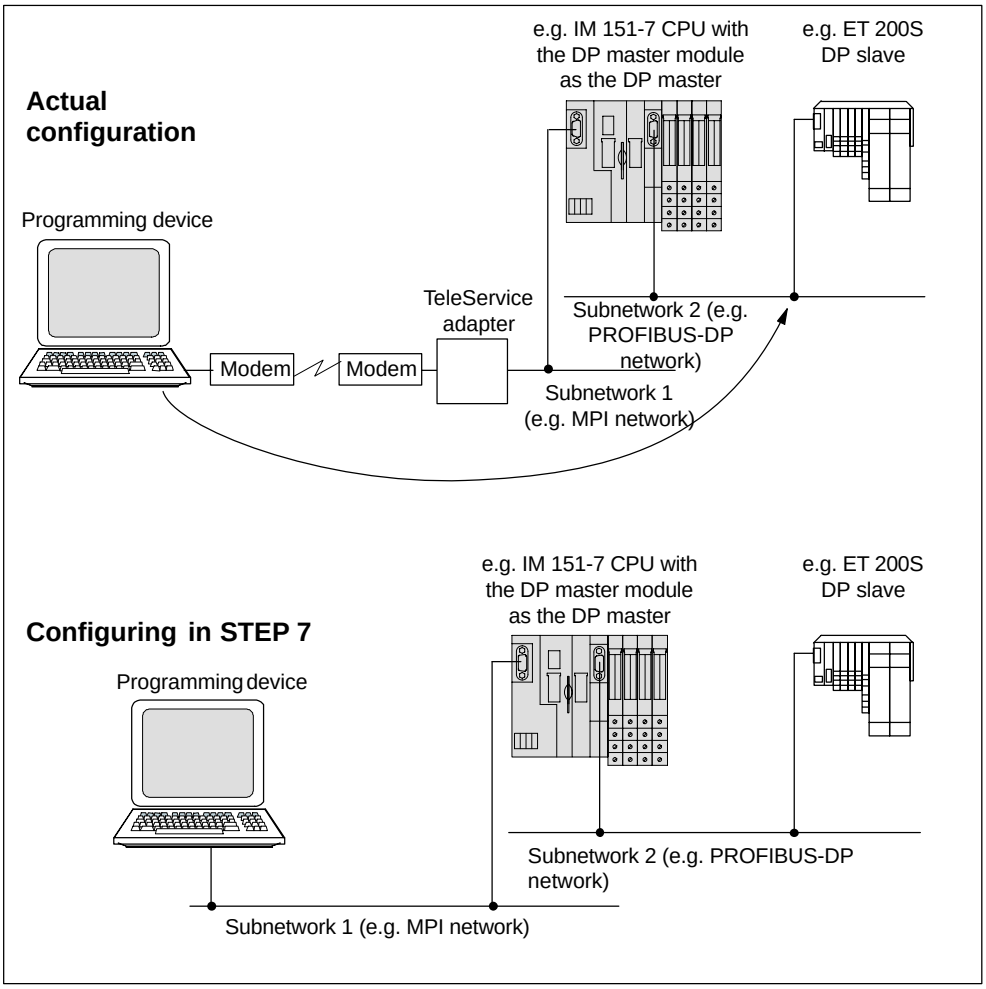


Figure 8-9 Routing – TeleService application example

More information ...

- More information on the configuration using *STEP 7* can be found in the *Configuring hardware and connections with STEP 7* manual.
- More basic information can be found in the *Communication with SIMATIC* manual.
- More information on the TeleService adapter can be found in the Internet. In the Manual Search section, you can find the relevant documentation to download under the A5E00078070 search term.

8.10 Data consistency

A data area is said to be consistent if it can be read/written by the operating system as a single block. The data that is transmitted together between devices should stem from one processing cycle and thus should form a unit, i.e. be consistent.

If there is a programmed communication function in the user program, for example X-SEND/ X-RCV, that accesses shared data, the access to this data area itself can be coordinated using the “BUSY“ parameter.

For PUT/GET functions

For S7 communication functions, such as PUT/GET and read/write via OP communication, that do not require a block in the user program of the CPU (as server), the size of the data consistency must be taken into account during programming.

The PUT/GET functions of the S7 communication or reading/writing of variables via the OP communication are processed in the cycle checking point of the CPU.

To ensure a defined process alarm reaction time, the communication variables are consistently copied in blocks of a maximum of 64 bytes in the cycle checkpoint of the operating system into and out of the user memory or memories. Data consistency is not guaranteed for any large data areas.

If a defined data consistency is required, the communication variables in the user program of the CPU may not exceed 64 bytes in size.

8.11 Blocks

This section provides an overview of the blocks that run in the IM 151-7 CPU.

The operating system is designed for event-driven processing of the user program. The following tables show which organization blocks (OBs) the operating system automatically invokes in response to which events.

More information

You will find a detailed description of the blocks in the *System and Standard Functions Reference Manual*. There is an overview of all the *STEP 7* documentation in the *ET 200S Distributed I/O System* manual.

Overview of all the blocks

Table 8-16 Overview of the blocks

Block	Number	Area	Maximum size	Remarks
OB	17	–	16 kB	All the possible OBs are found in the *Instruction list*.
FC	512	0-511		–
FB	512	0-511		–
DB	511	1-511		0 is reserved
SFC	61	–	–	All SFCs for the CPU are found in the *Instruction list*.
SFB	11	–	–	All SFBs for the CPU are found in the *Instruction list*.

A maximum of 1024 blocks (number of FBs + FCs + DBs) can be loaded in each IM 151-7 CPU.

SFC 55 ”WR_PARM”, SFC 56 ”WR_DPARM”, SFC 57 ”PARM_MOD”, SFC 58 ”WR_REC”

The use of SFCs 55 to 58 in conjunction with your IM 151-7 CPU is not recommended owing to the static module parameters.

If you do use SFCs 55 to 58 in conjunction with your IM 151-7 CPU, the IM 151-7 CPU could malfunction.

Points to note about OB 122

Note

Note the following about OB 122:

The CPU enters the value ”0” in the following temporary variables of the variable declaration table in the local data of the OB:

- **Byte No. 3:** OB122_BLK_TYPE
  (type of block in which the error occurred)
- **Byte No. 8 and 9:** OB122_BLK_NUM
  (number of block in which the error occurred)
- **Byte No. 10 and 11:** OB122_PRG_ADDR
  (address of block in which the error occurred)

8.12 Parameters

Parameterizable features

The properties and responses of the IM 151-7 CPU can be parameterized. You carry out this parameterization on different tabs in *STEP 7*.

Which parameters can be set for the IM 151-7 CPU?

The following table contains all the parameter blocks for the IM 151-7 CPU. The parameters are explained in the *STEP 7* online help.

Table 8-17 Parameter blocks, settable parameters and their ranges for the IM 151-7 CPU

Parameter blocks	Settable parameters	Range
Startup	Startup at preset configuration not equal to actual configuration	Yes/no
	Startup after power on	Warm restart
	Monitoring time for: • "Finished" message by means of modules (100 ms) • Transfer of the parameters to modules (100 ms)	 1 to 650 1 to 10000
Diagnostics/Clock	Report cause of STOP	Yes/no
	Synchronization in PLC • Type • Interval	 None/as master 1 s/10 s/1 min/10 min/1 h/12 h/24 h
	Synchronization to MPI • Type • Interval	 None/as master/as slave 1 s/10 s/1 min/10 min/1 h/12 h/24 h
	Correction factor	–10000 to +10000
Retentivity	Number of memory bytes starting with MB 0	0 to 255
	Number of S7 timers starting with T 0	0 to 255
	Number of S7 counters starting with C 0	0 to 255
Time-of-day interrupts	OB 10 activation	Yes/no
	OB 10 execution	• None • Once • Every minute • Hourly • Daily • Weekly • Monthly • Last day of the month • Annually
	Start date for the OB 10	Year-month-day
	Start time for the OB 10	Hours:minutes

Table 8-17 Parameter blocks, settable parameters and their ranges for the IM 151-7 CPU, continue

Parameter blocks	Settable parameters	Range
Cyclic interrupts	Periodicity of the OB 35 (ms)	1 to 60000
Cycle/clock memory	Scan cycle monitoring time (ms)	1 to 6000
	Cycle load from communication (%)	10 to 50
	OB85 call at I/O access error	• For each access • For incoming and outgoing errors • No call
	Clock memory	Yes/no
	Memory byte	0 to 255
Protection	Level of protection	• 1: Key switch • 2: Write protection • 3: Write/readprotection
	Mode	• Process mode: permissible cycle increase (ms) from 3 to 65535 • Test mode
Communication	PD communication	1 to 11
	OP communication	1 to 11
	S7 basic communication	0 to 10
Parameters	Interference frequency suppression	50 Hz / 60 Hz
	Bus length (see Note)	≤ 1 m / > 1 m
	Number of reference junctions • Activated • Slot • Channel number	1 • Yes/no • From 5 to 66 • 0/1

Note

The configuration length of an ET 200S station must not exceed 2 meters.

When does the CPU ”accept” the parameters?

The CPU accepts the parameters (configuration data) you have set:

- After POWER ON or a memory reset of the inserted memory module
- After the configuration data has been transferred without errors to the CPU online in STOP mode

8.13 Parameterization of the reference junction for the connection of thermocouples

If you want to use the IM 151-7 CPU in an ET 200S system with thermocouples and reference junctions, set the following parameters in the properties section of the hardware configuration:

Table 8-18 Parameterization of the reference junction

CPU parameter	Range	Explanation
Activation of the reference junction	Activated/not activated Example, see Figure 8-10	You can enable the reference junction with this parameter. Only then can you continue to parameterize the reference junction.
Slot	None/5 to 66 Example, see Figure 8-10	You can use this parameter to assign the RTD module slot to the reference junction.
Channel number	RTD on channel 0 RTD on channel 1 Example, see Figure 8-10	You can use this parameter to define the channel (0/1) for reference temperature measurement (determining the compensation value) for the assigned RTD module slot.
RTD module parameter	**Range**	**Explanation**
Measurement type/measurement range	Resistance/temperature measurement, e.g. • RTD-4L Pt 100 standard range	If you use a channel of the RTD module for reference junction parameterization, you must parameterize the measurement type/measurement range for this channel as *RTD-4L Pt 100 climatic range*.
TC module parameter	**Range**	**Explanation**
Reference junction number	1	This parameter allows you to assign the reference junction (1) that contains the reference temperature (compensation value).
Reference junction channel 0 and reference junction channel 1	None, RTD	You can enable the use of the reference junction with this parameter.

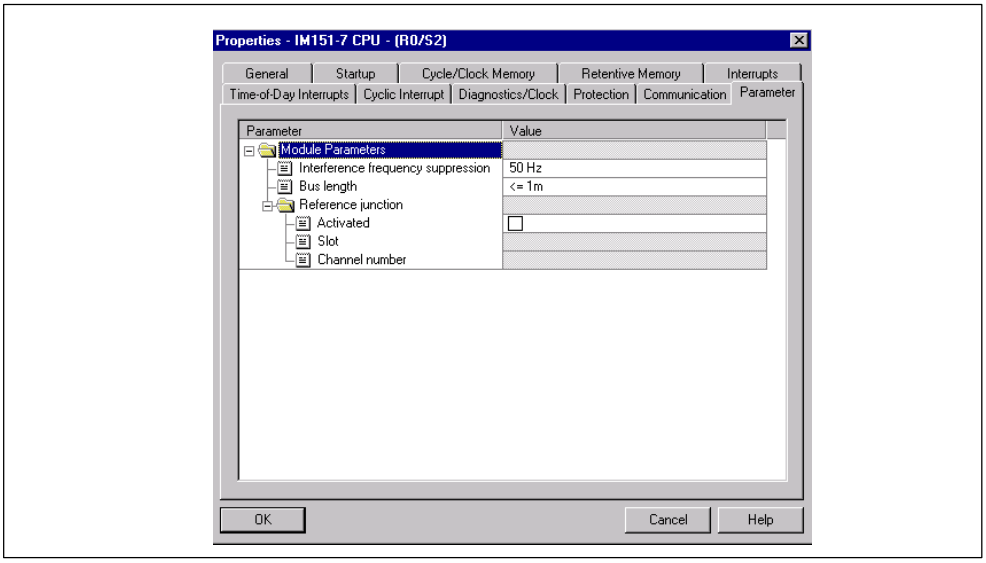


Figure 8-10 Example of a parameterization dialog box for the CPU module parameters in STEP 7 V5.2 + SP1

Reference

You can find detailed information on the procedure, the connection system and an example of parameterization in the chapter entitled *Analog Electronic Modules* in the *ET 200S Distributed I/O System* manual.

8.14 Removal and insertion of modules during operation

Removing and inserting one module at a time in the case of the IM 151-7 CPU with local ET 200S I/Os is possible during operation and in an energized state.

Exceptions

The CPU itself must not be removed during operation and in an energized state.

Removal and insertion of modules in an energized state and during operation

When removing and inserting modules in an energized state and during operation, refer both to the specifications in the manual and to the restrictions in the manual *ET 200S Distributed I/O System,* Section: *"Wiring and fitting"*.

Warning

When an output module is inserted, the outputs set by the user program become active immediately. We therefore advise you to set the outputs to "0" in the user program before removing the module.

If you do not remove and insert modules correctly (see the manual: *ET 200S Distributed I/O System,* Chapter: *"Wiring and fitting"*.) the system could malfunction. Adjacent modules could be affected.

What happens when modules are removed during operation

When you remove a module from the ET 200S I/O system during operation, OB 83 is called and a corresponding diagnostic buffer entry is created (event ID 3861_H), irrespective of whether the power module is switched on or off.

If OB 83 is in the memory of the CPU, the IM 151-7 CPU remains in RUN.

The absence of the module is noted in the system status list.

If the module that has been removed is accessed from the user program, an I/O access error occurs with a corresponding entry in the diagnostic buffer. In addition, OB 122 is called.

If OB 122 is in the memory of the CPU, the IM 151-7 CPU remains in RUN.

What happens when modules are inserted during operation

If you reinsert a module that has been removed in the ET 200S I/O system during operation, the CPU carries out a preset/actual comparison for the inserted module. This compares the configured module with the one that is actually inserted. The following activities occur depending on the result of the preset/actual comparison:

Modules that cannot be parameterized

The following activities occur irrespective of whether the power module of the inserted module is switched on or off.

Table 8-19 Result of the preset/actual comparison in modules that **cannot** be parameterized

Inserted module = configured module	Inserted module ≠ configured module
OB 83 is called with the corresponding diagnostic buffer entry (event ID 3861_H).	OB 83 is called with the corresponding diagnostic buffer entry (event ID 3863_H).
The module is entered in the system status list as available.	The module remains entered in the system status list as unavailable.
Direct access is again possible.	Direct access is not possible.

Modules that can be parameterized

The following activities only occur when the power module of the inserted module is switched **on**.

Table 8-20 Result of the preset/actual comparison in the case of parameterizable modules with the power module switched on

Inserted module = configured module	Inserted module ≠ configured module
OB 83 is called with the corresponding diagnostic buffer entry (event ID 3861_H).	OB 83 is called with the corresponding diagnostic buffer entry (event ID 3863_H).
The CPU reparameterizes the module.	The CPU does not parameterize the module.
If parameter assignment is successful, the module is entered in the system status list as available.	The module remains entered in the system status list as unavailable. The SF LED on the module remains lit.
Direct access is again possible.	Direct access is not possible.

The following activities only occur if the power module of the inserted module is switched **off**.

Table 8-21 Result of the preset/actual comparison in the case of parameterizable modules with the power module switched off

Inserted module = configured module	Inserted module ≠ configured module
OB 83 is called with the corresponding diagnostic buffer entry (event ID 3861_H).	
When the power module is switched on, the CPU reparameterizes the module.	When the power module is switched on, the CPU does not parameterize the module.
If parameter assignment is successful, the module is entered in the system status list as available.	The module remains entered in the system status list as unavailable. The SF LED on the module remains lit.
Direct access is again possible.	Direct access is not possible.

8.15 Switching power modules off and on during operation

What happens when power modules are switched off during operation

If the load power voltage to a power module is switched off during operation, the following activities take place:

- If you enable diagnostics during parameter assignment for the power module, diagnostic interrupt OB 82 (diagnostic address of the power module) is called with the corresponding diagnostic buffer entry (event 3942_H).
- The power module is entered as present but faulty in the system status list.

Switching off the load power supply has the following effects on the modules supplied by the power module:

- The SF LED on the modules lights up.
- The modules can continue to be accessed without an I/O access error occurring.
- The outputs of the modules are deenergized and inactive for the process.
- The inputs of digital modules and FM modules return 0; the inputs of analog modules return $7FFF_{H.}$

What happens when power modules are switched on during operation

If the load power supply to a power module is switched on during operation, the following activities take place:

- If you enable diagnostics when assigning parameters for the power module, the diagnostic interrupt OB 82 (diagnostic address of the power module) is called with the corresponding diagnostic buffer entry (event 3842_H).
- The power module is entered as present and o.k. in the system status list.

Switching on the load power supply has the following effects on modules supplied by the power module:

- The SF LED on the modules goes out.
- The modules regain their full functionality.

Removal and insertion of power modules during operation

You can find a list of activities that occur if you remove or insert power modules during operation in Section 8.14.

Removal and insertion has the same effects as switching the load power supply off and on for the modules that are supplied by the power module.

Cycle and Response Times

9

Introduction

In this chapter you will learn what make up the cycle and response times of the ET 200S with the IM 151-7 CPU.

You can use the PD to read out the cycle time of your user program (see the *STEP 7 user manual*).

The response time is more important for the process. In this chapter we will show you in detail how to calculate the response time.

Chapter overview

Execution times

- For the *STEP 7* instructions that can be processed by the CPUs can be found in the *Instruction list*.
- For the SFCs/SFBs integrated in the CPUs can be found in the *Instruction list*.

9.1 Cycle time

Cycle time – a definition

The cycle time is the time that the operating system requires to process a program cycle – i.e. an OB 1 cycle – as well as all the program sections and system activities interrupting this cycle.

This time is monitored.

Components of the cycle time

Factors	Remarks
Operating system execution time	See Table 9-1
Process image transfer time (PII and PIQ)	See Table 9-2
User program execution time	Is calculated from the execution times of the individual instructions and a CPU-specific factor (see Table 9-3).
Loading through interrupts	See Table 9-4

The following figure shows the components that make up the cycle time:

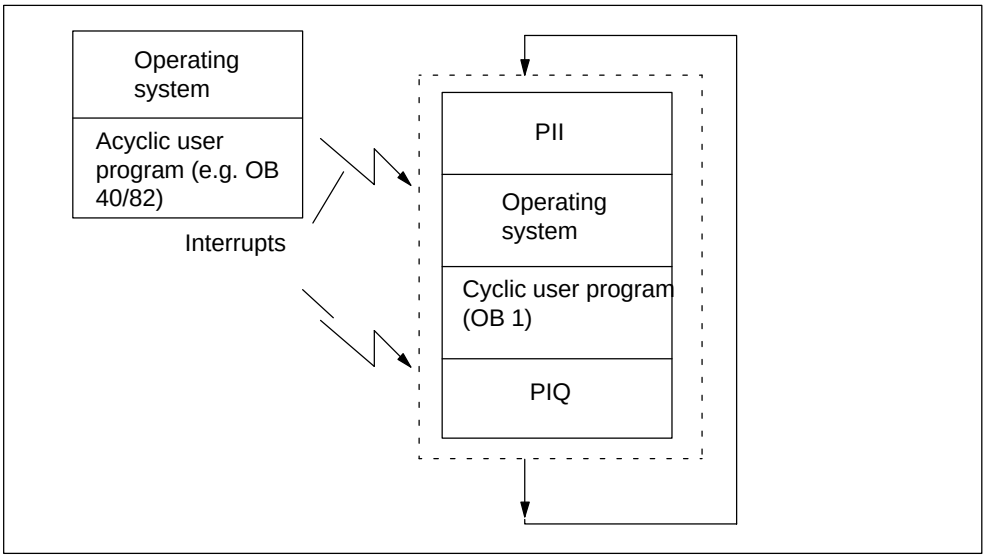


Figure 9-1 Component parts of the cycle time

Extending the cycle time

Note that the cycle time of a user program is extended by the following:

- Time-controlled interrupt handling
- Process interrupt handling (see also Section 9.3)
- Diagnostics and error handling (see also Section 9.3)

Operating system processing time

The operating system execution time for the IM 151-7 CPU is found in Table 9-1.

The specified time does not include the execution of:

- Testing and commissioning functions such as status/control of variables or block status
- Functions: Load block, delete block, compress block
- Communication
- Writing and reading the MMC with SFC 82 to 84

Table 9-1 Operating system processing time in the scan cycle checkpoint

Sequence	IM 151-7 CPU
Operating system processing time	600 µs

Process image transfer time

The table below contains the CPU times for process image updating (process image transfer time). The specified times can be longer as a result of interrupts that occur or by communication involving the IM 151-7 CPU.

(Process image = PI)

The CPU time for process image updating is calculated as follows:

K + A + D = Process image transfer time, in which K, A and D equal the following:

Table 9-2 Process image updating

	Designation	Times in the IM 151-7 CPU	
		Bus length ≤ 1 m	**Bus length > 1 m**
K	Base load	100 µs	150 µs
Q	Bytes in the PI for the ET 200S I/O	60 µs per byte	80 µs per byte
D	Per word in the DP area	1 µs	1 µs

User program scanning time:

The user program scanning time is made up of the sum of the execution times for the instructions and SFCs called. These execution times can be found in the instruction list. You also have to multiply the user program scanning time by an interface module-specific factor.

In the IM 151-7 CPU, this factor depends on the following:

Table 9-3 Dependency of the user program scanning time

Dependency	Range
The number of modules inserted	0 to 63

You can calculate the factor for your application approximately using the following rule of thumb for the IM 151-7 CPU:

	1.1
+	0.005 x number of modules
=	**Multiplier for your user program**

Delay of the inputs and outputs

You have to take into account the following delay times, depending on the expansion module:

- For digital inputs: The input delay time
- For digital outputs: Negligible delay times
- For analog inputs: Cycle time of the analog input
- For analog outputs: Response time of the analog output

Extending the cycle by nesting interrupts

Table 9-4 shows typical extensions of the cycle time through nesting of an interrupt. The program runtime at the interrupt level must be added to these. If several interrupts are nested, the corresponding times need to be added.

Table 9-4 Extending the cycle by nesting interrupts

Interrupts	IM 151-7 CPU
Process interrupt	500 µs
Diagnostic interrupt	600 µs
Time-of-day interrupt	400 µs
Delay interrupt	300 µs
Watchdog interrupt	150 µs
Status interrupt/update interrupt/manufacturer-specific interrupt	600 µs
Programming/access error/runtime system error	400 µs

9.2 Response time

Response time for the ET 200S with the IM 151-7 CPU

The response time is the time from the detection of an input signal at the ET 200S with the IM 151-7 CPU to the modification of an associated output signal via the inputs and outputs of the expansion modules.

Factors

The response time depends on the cycle time and the following factors:

Factors	Remarks
Delay of the inputs and outputs	You can find the delay times in the technical specifications of the electronic modules in the *ET 200S Distributed I/O System* manual.

Range of fluctuation

The actual response time lies between a shortest and a longest response time. You must always reckon on the longest response time when configuring your system.

The shortest and longest response times are considered below to let you get an idea of the width of fluctuation of the response time.

Shortest response time

The following figure shows you the conditions under which the shortest response time is obtained.

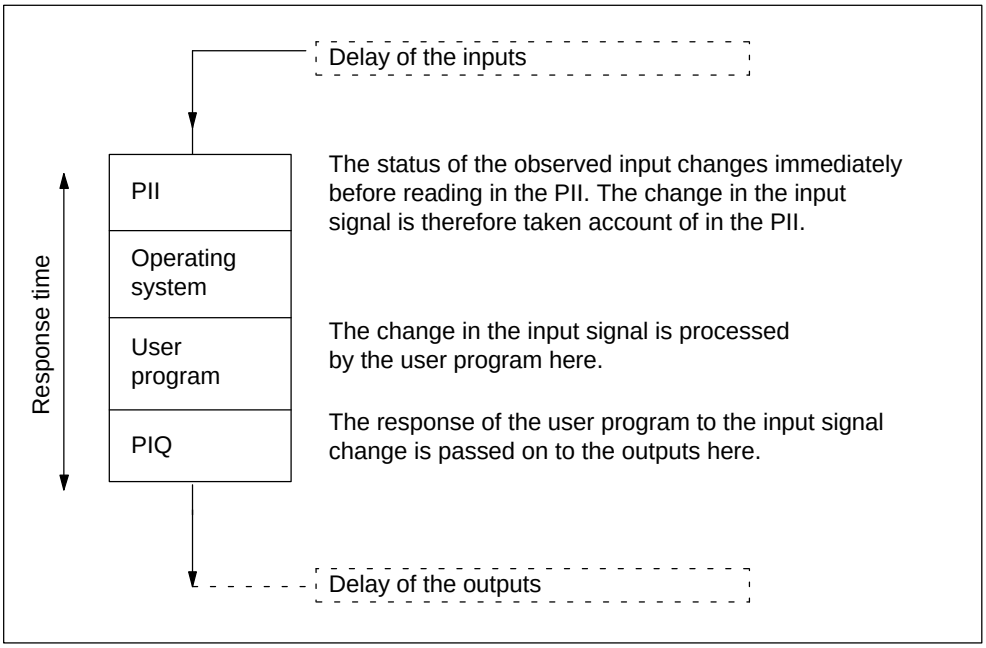


Figure 9-2 Shortest response time

Calculation

The (shortest) response time consists of the following:

- 1 × process image transfer time for the inputs +
- 1 × operating system processing time +
- 1 × program scanning time +
- 1 × process image transfer time for outputs +
- Delay of the inputs and outputs

This corresponds to the sum of the cycle time and the delay of the inputs and outputs.

Longest response time

The following figure shows what the longest response time consists of.

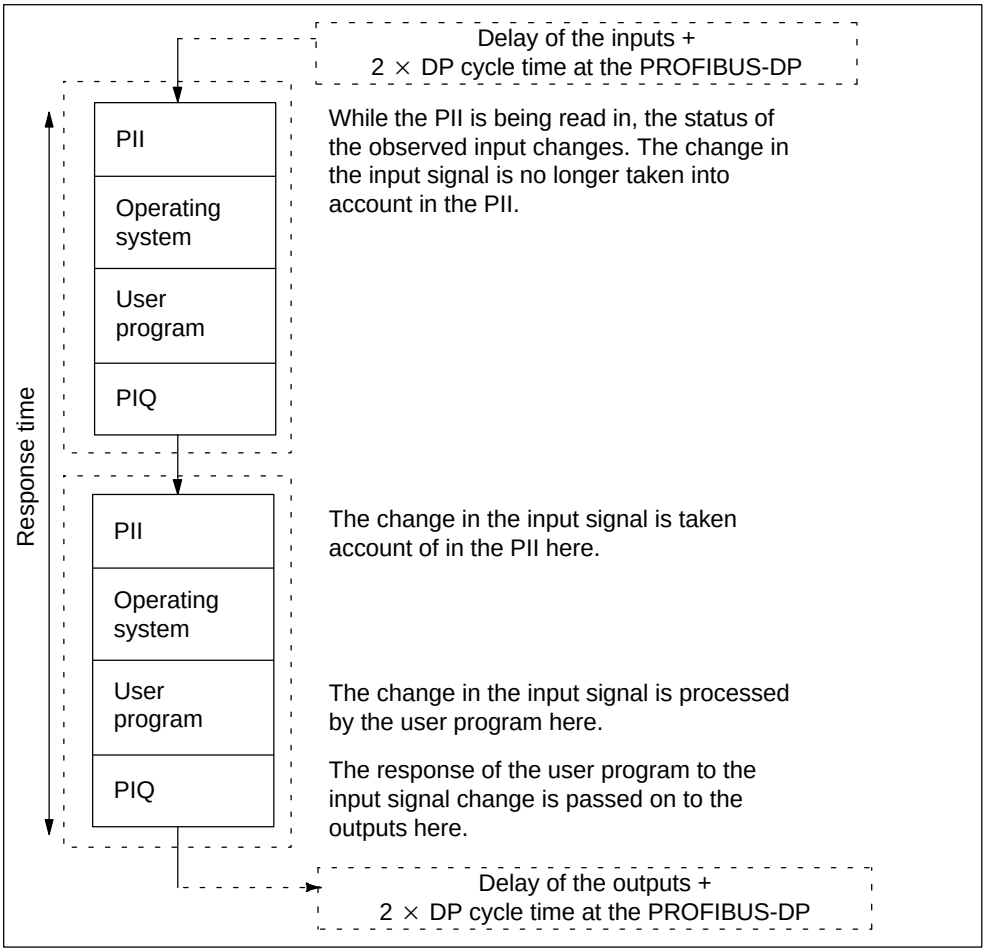


Figure 9-3 Longest response time

Calculation

The (longest) response time consists of the following:

- 2 × process image transfer time for the inputs +
- 2 × process image transfer time for the outputs +
- 2 × operating system processing time +
- 2 × program scanning time +
- 4 × run time of the DP slave frame (incl. processing in the DP master) +
- Delay of the inputs and outputs

This corresponds to the sum of 2x cycle time and delay of the inputs and outputs plus 4x DP cycle time.

9.3 Interrupt response time

Interrupt response time – a definition

The interrupt response time is the time from the first occurrence of an interrupt signal to the call of the first instruction in the interrupt OB of the IM 151-7 CPU.

The following generally applies: Interrupts of higher priority have precedence. This means that the program processing time of the higher-priority interrupt OB and of the not yet processed interrupt OBs of the same priority which occurred previously (queue) is added to the interrupt response time.

Interrupt response times

Table 9-5 Interrupt response times of the IM 151-7 CPU (without communication)

Interrupt response times (without communication) for...	Duration
Process interrupt, diagnostic interrupt	Less than 20 ms

Process interrupt handling

Process interrupt handling begins when the process interrupt OB 40 is called. Higher-priority interrupts cause the process interrupt handling routine to be interrupted. Direct accesses to the I/Os are made at the execution time of the instruction. After process interrupt handling has been completed, either cyclic program scanning is continued or additional interrupt OBs of the same priority or a lower priority are called and processed.

Technical Specifications

10

In this chapter

In this chapter you will find:

- The technical specifications of the IM 151-7 CPU interface module

10.1 Technical specifications of the IM 151-7 CPU

Order numbers

Interface module IM 151-7 CPU:	6ES7 151-7AA10-0AB0
DP master module:	6ES7 138-4HA00-0AB0
SIMATIC micro memory card (MMC): (see Section 8.3)	6ES7 953-8Lxx0-0AA0

Features

The IM 151-7 CPU interface module has the following features:

- Intelligent slave with RS 485 interface to the PROFIBUS-DP
- Stand-alone operation (MPI) possible
- 48 kByte working memory, non-expandable, retentive with inserted MMC
- Plug-in load memory on the MMC, up to 8 MBytes
- Powerfail-proof storage of the user program and configuration via MMC
- Configurable with *STEP 7*, as of V5.1 + Service Pack 4
- Maximum configuration of the local I/Os:
  - 63 ET 200S modules
  - Bus length of 2 meters

In addition to the features listed above, the IM 151-7 CPU interface module with the DP master module has DP master functionality. Up to 32 DP slaves can be connected to the DP master interface. You will require *STEP 7* of V5.2 + SP1 or higher to configure the system.

General technical specifications

The IM 151-7 CPU / DP master module meets the general technical specifications of the ET 200S distributed I/O system. You will find these standards and test specifications in the chapter entitled ”General technical specifications” in the *ET 200S Distributed I/O System* manual.

Terminal assignment for the IM 151-7 CPU

Table 10-1 Terminal assignment for the interface module IM 151-7 CPU

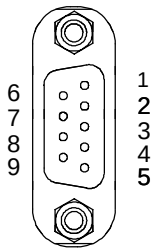
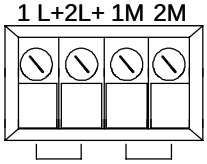
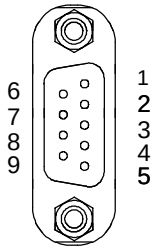
View		Signal name	Description
IM 151-7 CPU  RS 485 interface	1	–	–
	2	M24	External 24 VDC supply
	3	RxD/TxD-P	Data line B
	4	RTS	Request To Send
	5	M5V2	Data reference potential (from the station)
	6	P5V2	Supply plus (from the station)
	7	P24	External 24 VDC supply
	8	RxD/TxD-N	Data line A
	9	–	–
1 L+2L+ 1M 2M 	1 L+		24 VDC
	2L+		24 VDC (to loop through)
	1M		Masse
	2M		Chassis ground (to loop through)

Table 10-2 Pin assignment of the DP master module

View		Signal name	Description
DP master module  RS 485 interface	1	–	–
	2	–	–
	3	RxD/TxD-P	Data line B
	4	RTS	Request To Send
	5	M5V2	Data reference potential (from the station)
	6	P5V2	Supply plus (from the station)
	7	–	–
	8	RxD/TxD-N	Data line A
	9	–	–

Basic circuit diagram for the IM 151-7 CPU

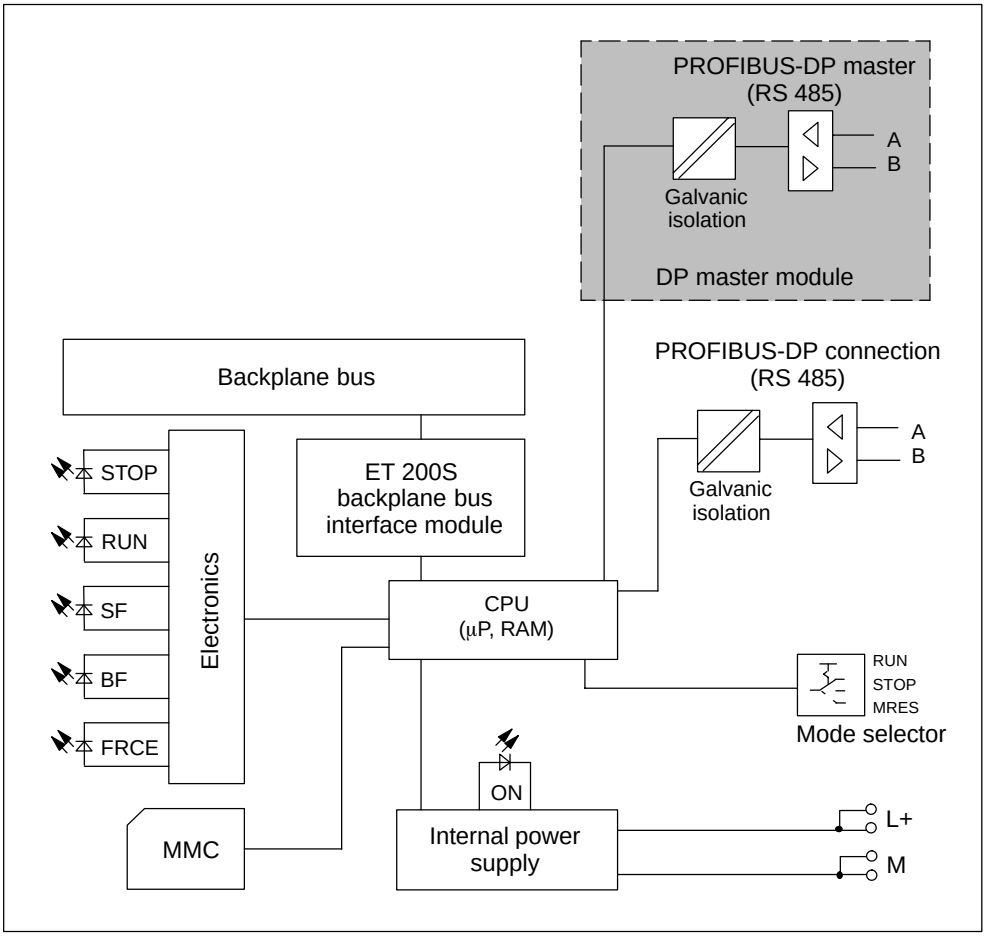


Figure 10-1 Basic circuit diagram for the IM 151-7 CPU

Technical specifications

CPU and product version	
MLFB	6ES7 151-7AA10-0AB0
• Hardware version	02
• Firmware version	V2.1.0
• Matching programming package	• STEP 7 starting with V5.1 + SP6 (for IM 151-7 CPU) Optional: – S7-SCL – S7-GRAPH • STEP 7 as of V5.2 + SP1 (for IM 151-7 CPU with the DP master module)
Memory	
Working memory	
• Integral	48 kBytes
• Expandable	No
Load memory:	Plug-in (MMC up to 8 MB)
• Data storage on the MMC (after the last programming)	At least 10 years
Backup	Guaranteed by MMC (maintenance-free)
Processing times	
Processing times for	
• Bit instructions	Min. 0.1 µs
• Word instructions	Min. 0.2 µs
• Fixed-point math instructions	Min. 2 µs
• Floating-point math instructions	Min. 6 µs
Timers, counters and their retentive features	
S7 counters	256
• Retentivity	Adjustable
• Preset	From C 0 to C 7
• Counting range	0 to 999
IEC counters	Yes
• Number	Unlimited (limited only by the working memory)
• Type	SFB
S7 timers	256
• Retentivity	Adjustable
• Preset	No retentive timers
• Timing range	10 ms to 9990 s
IEC timers	Yes
• Number	Unlimited (limited only by the working memory)
• Type	SFB
Data areas and their retentive features	
Total retentive data area (incl. memory markers, timers, counters)	All
Memory markers	256 bytes
• Retentivity	Adjustable
• Preset	MB 0 to MB 15
Clock memory	8 (1 memory byte)
Data blocks	Max. 511 (DB 0 reserved)
• Size	Max. 16 kByte
Local data per priority class	Max. 510 bytes
Blocks	
Total	1024 (FBs + FCs + DBs)
FBs	Max. 512
• Size	Max. 16 kBytes
FCs	Max. 512
• Size	Max. 16 kBytes
DBs	Max. 511
• Size	Max. 16 kBytes
OBs	See *Instruction list*
• Size	Max. 16 kByte
Nesting depth:	
• Per priority class	8
• Additional levels within an error OB	4
Address areas (inputs/outputs)	
Total I/O address area	Max. 2048 bytes/2048 bytes
Process image	128 bytes/128 bytes (not adjustable)
Digital channels	Max. 16336/16336
• Central	248/248
Analog channels	Max. 1021/1021
• Central	124/124

Configuration rules	
• Max. 63 I/O modules per station	
• Station width <1 m or <2 m	
• Max. 10 A per load group (power module)	
• Master interface module, right, near the IM 151-7 CPU (X2 interface)	
Time	
Clock	Hardware clock
• Buffered	Yes
• Backup time	Typ. 6 weeks (at ambient temperature of 40 °C)
• Accuracy	Deviation per day < 10 s
Run-time meter	1
• Number	0
• Range	0 to 2^{31} hours (when using the SFC 101)
• Selectivity	1 hour
• Retentive	Yes; must be restarted with every restart
Time synchronization	Yes
• In the PLC	No
• On the MPI	Master/slave
S7 message functions	
Number of logon-able stations for signaling functions (e.g. OS)	Max. 12 (depending upon the configurations configured for PG/OP and S7 basis communication)
Process diagnostic messages	ALARM_S, ALARM_SC, ALARM_SQ
• Simultaneously active Alarm_S blocks	Max. 40
Testing and commissioning functions	
Status/modify variables	Yes
• Variable	Inputs, outputs, memory markers, DBs, timers, counters
• Number of variables	Max. 30
– Of which status variables	Max. 30
– Of which modify variables	Max. 14
Force	Yes
• Variable	Inputs, outputs
• Number	Max. 10
Monitor block	Yes
Single sequence	Yes
Breakpoint	2
Diagnostic buffer	Yes
• Number of inputs	Max. 100 (not adjustable)
Communication functions	
PD/OP communication	Yes
Global data communication	Yes
• Number of GD packets	Max. 4
– Sender	Max. 4
– Receiver	Max. 4
• Size of GD packets	Max. 22 bytes
– Number of which consistent	22 bytes
S7 basic communication	Yes
• User data per job	Max. 76 bytes
– Number of which consistent	76 bytes (XSEND/XRCV) 64 bytes (XPUT/XGET) as server
S7 communication	Yes (server)
• User data per job	Max. 180 bytes
– Number of which consistent	64 bytes
S5-compatible communication	No
Standard communication	No
Number of connections	Max. 12
Used for	
• PD communication	Max. 11
– Reserved (default)	1
• OP communication	Max. 11
– Reserved (default)	1
• S7 basic communication	Max. 10
– Reserved (default)	0
Routing	Max. 4

	• As slave only when the interface is active
	• With the IM 151-7 CPU as the DP master
Interfaces	
On IM 151-7 CPU (X1)	
Type of interface	Integrated RS 485 interface
Physical system	RS 485
Galvanically isolated	Yes
Power supply to the interface (15 to 30 V DC)	Max. 80 mA
Functionality	
• MPI	Yes
• PROFIBUS-DP	DP slave (active/passive)
• Point-to-point connection	No
MPI	
• Number of connections	12**
• Utilities:	
– PD/OP communication	Yes
– Routing	Yes (with master module)
– Global data communication	Yes
– S7 basic communication	Yes
– S7 communication	Yes (only server)
• Transmission rates	Max. 12 Mbaud
DP slave	
• Number of connections	12**
• Utilities:	
– PD/OP communication	Yes
– Routing	Yes (only with active interface and in the master operating mode)
– Direct communication	Yes
• Transmission rates	Up to 12 Mbaud
• Automatic transmission rate search	Yes (only with passive interface)
• Intermediate memory	244 I bytes/244 O bytes
– Address areas	32 with a maximum of 32 bytes each *
• DPV1	No
• Device master file	You can find the current device master file at `http://www.ad.siemens.de/csi_e/gsd.`
On the DP master module (X2)	
Type of interface	External interface through the master module 6ES7138-4HA00-0AB0)
Physical system	RS 485
Galvanically isolated	Yes
Power supply to the interface (15 to 30 VDC)	No
Functionality	
• MPI	No
• PROFIBUS-DP	DP master
• Point-to-point connection	No
DP master	
• Number of connections	12**
• Utilities:	
– PD/OP communication	Yes
– Routing	Yes
– Global data communication	No
– S7 basic communication	No
– S7 communication	Yes (only server)
– Direct communication	Yes
– Clock synchronism	Yes
– SYNC/FREEZE	Yes
– Activate/deactivate DP slaves	Yes
– DPV1	Yes

• Transmission rates	Up to 12 Mbaud
• Number of DP slaves per station	32
• Address area	Max. 2 kbytes I/ 2 kbytes O
• User data per DP slave	Max. 244 bytes I/ 244 bytes O
Programming	
Programming language	STEP 7 (LAD, FBD, STL)
Stored instructions	See *Instruction list*
Nesting levels	8
System functions (SFCs)	See *Instruction list*
System function blocks (SFBs)	See *Instruction list*
User program security	Yes
Dimensions and weight	
Installation dimensions W × H × D (mm)	
• IM 151-7 CPU	60 x 119.5 x 75
• DP master module	35 x 119.5 x 75
Weight	
• IM 151-7 CPU	Approx. 200 g
• DP master module	Approx. 100 g
Voltages, currents, potentials	
Rated supply voltage	24 VDC
• Permissible range	20.4 to 28.8 V
• Polarity reversal protection	Yes
• Short-circuit protection	Yes
• Voltage failure buffering	5 ms

Galvanic isolation	
• Between the supply voltage (1L+) and all other circuit components	Yes
• Between PROFIBUS-DP and all other circuit components	Yes
• Between the supply voltage (1L+) and PROFIBUS-DP	Yes
• Between the PROFIBUS-DP slave and PROFIBUS-DP master	Yes
Permitted potential difference	
• Between different circuits	75 VDC, 60 VAC
Insulation tested at	500 VDC
Current consumption from supply voltage (1L+)	
• IM 151-7 CPU	Approx. 250 mA
• IM 151-7 CPU + DP master module	Approx. 280 mA
• Power supply for the ET 200S backplane bus	Max. 700 mA
Power losses	Typ. 3.3 W

* Up to the maximum size of the intermediate memory

** Attention: 12 connections per CPU, not per interface.

Changing IM 151-7 CPU (6ES7 151-7Ax00..) to IM 151-7 CPU (6ES7 151-7AA10-0AB0) 11

If you download your existing user program for the IM 151-7 CPU (6ES7 151-7Ax00-0AB0) to an IM 151-7 CPU (6ES7 151-7AA10-0AB0), you may encounter the following problems:

The SFC 56, SFC 57 and SFC 13 may work asynchronously

A number of asynchronous SFCs are always processed or, under certain conditions, already processed following the first call (”quasi-synchronous”) on the IM 151-7 CPUs (6ES7 151-7Ax00-0AB0).

These SFCs actually run asynchronously on the IM 151-7 CPUs (6ES7 151-7AA10-0AB0). The asynchronous processing may continue for several OB 1 cycles. As a result, a queue within an OB may become a continuous loop.

- SFC 56 ”WR_DPARM”; SFC 57 ”PARM_MOD”

  This SFCs always function ”quasi-synchronously” on stand-alone IM 151-7 CPUs (6ES7 151-7Ax00-0AB0).
  They function asynchronously on stand-alone IM 151-7 CPUs (6ES7 151-7AA10-0AB0) and distributed IM 151-7 CPUs.

Note

If you use the SFC 56 ”WR_DPARM” or SFC 57 ”PARM_MOD”, you should always evaluate the BUSY bit of the SFCs.

- SFC 13 ”DPNRM_DG”

  This SFC always runs “quasi-synchronously” when called in OB 82.
  It always runs asynchronously on the IM 151-7 CPU (6ES7 151-7AA10-0AB0).

Note

In the user program, all that should take place is that the task should be called in the OB 82. The evaluation of the data, which should take into account the BUSY bits and the acknowledgement in the RET_VAL, should take place in the cyclic program.

Tip:
We recommend using the SFB 54 instead of the SFC 13 with the IM 151-7 CPU (6ES7 151-7AA10-0AB0).

SFC 20 ”BLKMOV”

Previously this SFC could also be used to copy data from a non-process-related DB.

The SFC 20 no longer has this functionality. The SFC 83 ”READ_DBL” must now be used for this purpose.

SFC 54 ”RD_DPARM”

This SFC is no longer available. The asynchronous SFC 102 ”RD_DPARA” must be used instead.

SFCs which may give different results

If you only use logical addressing in your user program, you can ignore the following points.

If you use address conversions in the user program (SFC 5 ”GADR_LGC”, SFC 49 ”LGC_GADR”), you must check the slot assignment and logical start address assignment for DP slaves.

- The diagnostic address of the DP slave is now always assigned to slot 0 (station proxy).
- The DP slave is integrated in *STEP 7*:

  The interface module (slot 2) may have its own address
  (e.g. IM 151-7 CPU as I slave).

Changed runtimes during program processing

If you have created a user program that is optimized for the execution of certain processing times, it is important to note the following when using the IM 151-7 CPU (6ES7 151-7AA10-0AB0):

- Program processing in the IM 151-7 CPU is significantly faster.
- Functions that require MMC access (e.g. system runup, program download in RUN, DP station recovery, etc.), may run slower on the IM 151-7 CPU.

Changing the diagnostic addresses of DP slaves

It is important to note that the diagnostic addresses for the slaves may have to be reassigned in some cases since two diagnostic addresses per slave are sometimes required due to adaptation to the DPV1 standard.

- The virtual slot 0 has its own address (diagnostic address of the station proxy).

  The module state data for this slot (read out with SFC 51 ”RDSYSST”) contains the identifiers which affect the complete slave/station, e.g. the ”Station Faulty” identifier. The station failure and station recovery are also signaled in the OB 86 of the master via the diagnostic address of the virtual slot 0.
- Furthermore, slot 2 also has its own address in the case of modules integrated in *STEP 7* (e.g. IM 151-7 CPU as I slave). This address is used to signal the change in operating state in the diagnostic interrupt OB 82 of the master, e.g. with IM 151-7 CPU as I slave.

Note

Reading out the diagnosis with SFC 13 “DPNRM_DG”:
The diagnostic address originally assigned still functions. Internally, *STEP 7* assigns slot 0 to this address.

If you use the SFC 51 ”RDSYSST” to read out, for example, module state information or rack/station state information, you must also take the changed meaning of the slots and the additional slot 0 into consideration.

Using consistent data areas in the process image with DP slaves

Below is a list of the points to which you must pay attention with communication in a DP master system if you want to transfer I/O areas with the ”Total Length” consistency.

- If the address area of consistent data is in the process image, this area is updated automatically.
- You can also use the SFCs 14 and 15 to read and write consistent data.
- If the address area for consistent data is outside the process image, you must use the SFCs 14 and 15 to read and write consistent data.
- It is also possible to address the consistent areas directly (for example, L PIW or T PQW).

You can transfer a maximum of 32 bytes of consistent data.

Replacing an IM 151-7 CPU (6ES7 151-7Ax00-0AB0) with an IM 151-7 CPU (6ES7 151-7AA10-0AB0) in the configuration

If the user does not make any changes to the configuration, the functional settings in the configuration are set to default values if an IM 151-7 CPU (6ES7 151-7Ax00-0AB0) is replaced with an IM 151-7 CPU (6ES7 151-7AA10-0AB0).

This has the following effect:

- The IM 151-7 CPU (6ES7 151-7Ax00-0AB0) was set to ”No DP” (i.e. stand-alone).
  → the IM 151-7 CPU (6ES7 151-7AA10-0AB0) is set to ”MPI”.
- The IM 151-7 CPU (6ES7 151-7Ax00-0AB0) was set to ”DP Slave”.
  → the IM 151-7 CPU (6ES7 151-7AA10-0AB0) is also set to ”DP Slave”.

Information for replacement in HWConfig

- Marking and then replacing the IM 151-7 CPU does not function.
- Replacement is only possible after the rack has been selected.

PD/OP functions

In IM 151-7 CPU (6ES7 151-7Ax00-0AB0), PD/OP functions at the DP interface were only possible at an active interface.

In IM 151-7 CPU (6ES7 151-7AA10-0AB0), these functions are possible at both passive and active interfaces. However, the performance at the passive interface is noticeably lower.
In IM 151-7 CPU with a DP master module, the PD/OP functions are also available via the DP master interface.

Routing in IM 151-7 CPU used as an I slave

If you are using the IM 151-7 CPU (6ES7 151-7AA10-0AB0) as an intelligent slave, the routing function can only be used when the DP interface is active.
Activate the Commissioning/Test Mode checkbox in the DP interface properties in STEP 7.

New functionality of the IM 151-7 CPU (6ES7 151-7AA10-0AB0)

- Coexistent MPI/DP interface (active/passive) (see Section 8.5)
- DP master interface in combination with the DP master module (see Chapters 6.1 and 8.5)
- New memory concept (see Section 8.4)
- Global data communication (see Section 8.8)

  This utility is used to enable the cyclic exchange of global data between SIMATIC S7-CPUs (and therefore also the IM 151-7 CPU).
- S7 basic communication (see Section 8.8)

  This utility is used to enable data exchange between the IM 151-7 CPU and communications-capable SIMATIC modules within an S7 station. The SFCs 65 to 74 are provided to support this.
- MMC up to 8 MBytes (see Section 8.3)
- Data storage (see Section 8.4.4)

  The data is stored on an MMC and then downloaded back to the CPU using the SFCs 82 to 84.
- Storage of the STEP 7 project on an MMC (see Section 8.4.5)
- New SFBs

  The SFBs 52 to 54 and SFB 75 to IEC 61784-1 are supported.
- 32-bit run-time meter

  The meter is operated using the SFC 101.
- Routing (with DM master module)
- Non-retentive DBs (created using SFC 82 or in *STEP 7*) (see Chapter 8.4.1)

Master functionality of the IM 151-7 CPU with the DP master module

Activation/deactivation of DP slaves via the SFC 12
Slaves that were deactivated via the SFC 12 are automatically activated in a restart for the IM 151-7 CPU (transition from STOP to RUN).

Interrupt events from the distributed I/O while the CPU is in STOP mode
Due to the new DPV1 functionalities (IEC 61784-1:2002 Ed1 CP 3/1), how incoming interrupt events from the distributed I/O while the CPU is in STOP mode are handled also changes.
In IM 151-7 CPU, an interrupt event (process, diagnostic interrupt, new DPV1 interrupt) from the distributed I/O while the CPU is in STOP mode is already acknowledged and, if applicable, entered in the diagnostic buffer (diagnostic interrupt only). In the subsequent change of the CPU into RUN mode, the interrupt is no longer processed via the corresponding OB. Possible faults in slaves can be read out via corresponding SZL information (e.g. SZL 0x692 read out via SFC 51).

Position of the IM 151-7 CPU in the CPU Range

12

In this chapter, you will find out the most important differences to two selected CPUs in the S7-300 SIMATIC family.

We will also show you how to rewrite programs you have written for the S7-300-CPUs for the IM 151-7 CPU.

Chapter overview

More information

You can find further information on how to create and structure programs in the *STEP 7* manuals and online help.

12.1 Differences to selected S7-300 CPUs

The following table lists the most important programming differences between two selected CPUs of the S7-300 SIMATIC family and the IM 151-7 CPU.

Table 12-1 Differences to selected S7-300 CPUs

Features	CPU 315-2 DP	CPU 315-2 DP (modular)	IM 151-7 CPU	
			(6ES7 151-7Ax00-0AB0)	(6ES7 151-7AA10-0AB0)
Real-time clock	Hardware	Hardware	Software	Hardware
Memory backup	Yes, with battery	Guaranteed by MMC (maintenance-free)	Not possible	Guaranteed by MMC (maintenance-free)
Memory card	Memory card	MMC	MMC	MMC
Number of connections to PD and OP	4 (as of 10/99: 12)	16	5	Max. 12
Setting the PROFIBUS address	Hardware configuration	Hardware configuration	Hardware configuration must match address setter	Hardware configuration
Transmission rate to PD and OP	187.5 kbaud (MPI) 12 Mbaud (DP)	187.5 kbaud (MPI) 12 Mbaud (DP)	12 Mbaud (DP)	12 Mbaud (MPI/DP)
Communication:				
PD/OP	Yes	Yes	Yes	Yes
Global data comm.	Yes	Yes	No	Yes
S7 basic comm.	Yes	Yes	Yes (server)	Yes
S7 comm.	Yes (server)	Yes (server)	Yes (server)	Yes (server)
Direct communication	Yes	Yes	Yes	Yes
Range of uses with DP	As a DP master As a DP slave Stand-alone	As a DP master As a DP slave Stand-alone	As a DP slave Stand-alone	As a DP slave Stand-alone As the DP master (with the DP master module)
Addressing	Free	Free	Free	Free
Interrupt response time	0.4-1.3 ms	0.3-1.2 ms	Less than 20 ms	Less than 20 ms
Removal/insertion of modules during operation	No	No	Yes	Yes

12.2 Porting the user program

Introduction

By porting we mean making available on a distributed basis a program that was previously used centrally on a master. Certain adjustments may be necessary to relocate an existing program partially or completely from a master to an intelligent slave. The resources required for porting sections of a user program to an intelligent slave depend on how the address assignment of inputs and outputs is stored in the FBs in the source program.

The inputs and outputs can be used in the FCs in the source program in different ways. Addresses can be packed in the current ET 200S, which is not possible in the IM 151-7 CPU.

See the description of IM 151-7 CPU addressing in Section 3.1.

Porting with unpacked addresses

If you use FBs with unpacked I/O addresses, the required program sections can be transferred easily to the IM 151-7 CPU without the need for porting.

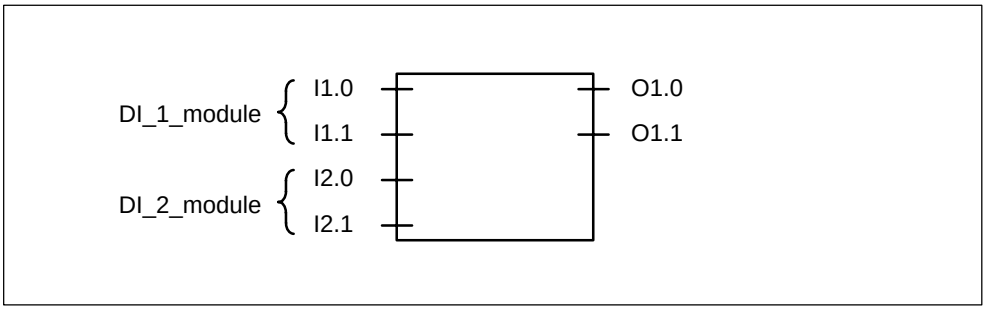


Figure 12-1 Example: FB with unpacked addresses

Porting with packed addresses

If FBs with packed I/O addresses are copied to the IM 151-7 CPU, the packed addresses there can no longer be assigned to the I/Os of the I/O modules locally because the CPU of the IM 151-7 CPU cannot work with packed addresses. This requires rewiring of the corresponding FBs. Rewiring corresponds to ”unpacking” the addresses.

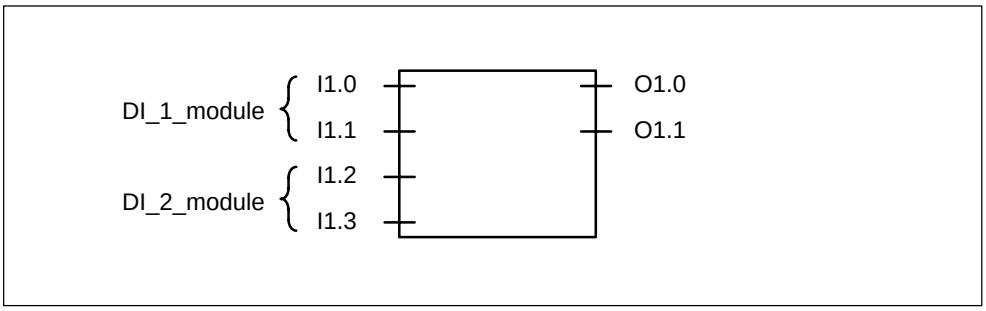


Figure 12-2 Example: FB with packed addresses

Rewiring

The following blocks and address IDs can be rewired:

- Inputs, outputs
- Memory markers, timers, counters
- Functions, function blocks

Proceed as follows to rewire the signals:

1. In SIMATIC Manager, select the ”Blocks” folder, which contains the blocks with the packed addresses that you want to port to the IM 151-7 CPU.
2. Select the menu command **Tools → Rewire**.
3. Enter the desired replacements in the displayed ”Rewiring” dialog box in the table (old address ID/new address ID).

Table 12-2 Example : Replacements under Tools → Rewire

	Old address ID	New address ID
1	I 1.2	I 2.0
2	I 1.3	I 2.1

4. Click OK.

This starts rewiring. After rewiring, you can decide in a dialog box whether you want to look at the rewiring information file. The file contains the list of old and new address IDs. The various blocks are also listed together with the number of rewirings carried out in the block.

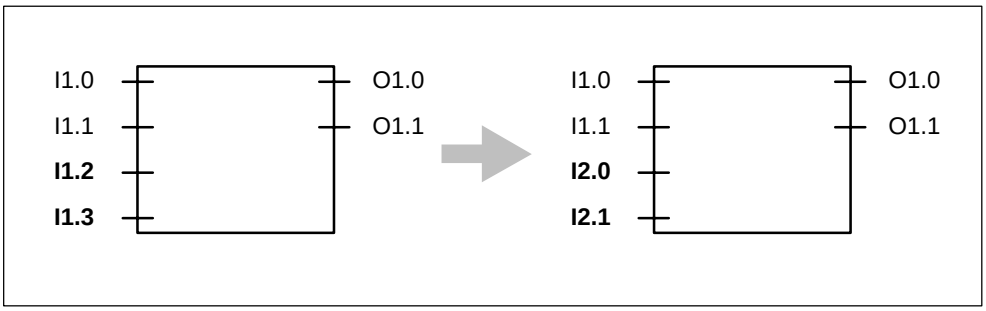


Figure 12-3 Example: Rewiring the signals

If you assign symbols to the inputs and outputs using the symbol table in *STEP 7*, you must change the symbol table to adjust the subprogram for use in the IM 151-7 CPU.

See also the *STEP 7* online help system.

Porting FBs with I/Os in an I/O word

If you map the addresses of the inputs and outputs via an I/O word to a function block you have programmed, porting is considerably more involved.

It is possible to program a shell around the FB that makes an adjustment so that the FB can be used with the IM 151-7 CPU. The other option is to reprogram the FB. We recommend you reprogram the FB because this is easier than programming a shell.

See also the *STEP 7* online help system.

Glossary

Address

An address is the designation for a certain address ID or address area (e.g. input I 12.1; memory word MW 25; data block DB 3).

AKKU

The accumulators are registers in the → CPU that act as an intermediate memory for loading and transfer operations as well as comparison, calculation and conversion operations.

Automation System

An automation system is a programmable logic controller that consists of at least one CPU, various input and output modules and human-machine interfaces.

Backup Memory

The backup memory ensures that the memory areas of the → CPU that do not have a buffer battery are buffered. A parameterizable number of times, counters, markers and data bytes is buffered.

Bus

A bus is the common transmission route to which all nodes are connected; has two defined ends.

In the case of the ET 200S, the bus is a two-wire or fiber-optic cable.

Compression

The programming device online function Compress is used to align all valid blocks contiguously in the RAM of the CPU at the start of the user memory. This eliminates gaps that occur when blocks are deleted or corrected.

Consistent Data

Data that belongs together with respect to its content and that must not be separated is referred to as consistent data.

For example, the values from analog modules must always be handled consistently, i.e. the value of an analog module must not be corrupted by reading it out at two different points in time.

Counter

Counters are part of the → system memory of the CPU. The content of the "counter cells" can be modified by *STEP 7* instructions (e.g. count up/down).

CPU

Central processing unit of the S7 programmable controller with a control unit and arithmetic logic unit, memory, operating system and interface for a programming device.

Cross communication

See Direct communication

Cycle Time

The cycle time is the time taken by the → CPU to scan the → user program once.

Data Block

Data blocks (DB) are data areas in the user program that contain user data. There are global data blocks that can be accessed from all code blocks and there are instance data blocks that are assigned to a specific FB call.

Device master file

A device master file contains all the DP slave-specific properties. The structure of the device master file (DDB) is defined in IEC 61784-1:2002 Ed1 CP 3/1.

Diagnosis

Diagnosis is the detection, localization, classification, display and further evaluation of errors, faults and messages.

Diagnosis offers monitoring functions that are executed automatically while the system is running. This increases the availability of the systems by reducing maintenance and standstill periods.

Diagnostic buffer

The diagnostic buffer is a buffered memory area in the CPU in which diagnostic events are stored in the order of their occurrence.

Diagnostic interrupt

Diagnostics-capable modules use diagnostic interrupts to report system errors which they have detected to the central CPU.

In SIMATIC S7/M7: When an error is detected or disappears (e.g. wire break), the ET 200S triggers a diagnostic interrupt, provided the interrupt is enabled. The CPU of the DP master interrupts the execution of the user program or lower-priority priority classes and processes the diagnostic interrupt block (OB 82).

In SIMATIC S5: The diagnostic interrupt appears in the station diagnosis. Using cyclical querying of the diagnostic bits in the station diagnosis you can detect errors such as a wire break.

Direct Communication

Direct communication is a special communication relationship between PROFIBUS-DP nodes. Direct communication is characterized by the fact that the PROFIBUS-DP nodes ”listen in” to find out which data a DP slave is sending back to its DP master.

Distributed I/O Systems

Distributed I/O devices are input/output devices that are not used in the central unit but set up at distributed locations at large distances from the CPU, e.g.:

- ET 200S, ET 200M, ET 200B, ET 200C, ET 200U, ET 200X, ET 200L
- DP/AS-I Link
- S5-95U with a PROFIBUS-DP slave interface
- Other DP slaves from either Siemens or other vendors

The distributed I/O systems are connected to the DP master via PROFIBUS-DP.

DP Master

A → master that behaves according to the IEC 61784-1:2002 Ed1 CP 3/1 standard is designated a DP master.

DP Slave

A → slave on the PROFIBUS bus system with the PROFIBUS-DP protocol that complies with IEC 61784-1:2002 Ed1 CP 3/1 is referred to as a DP slave.

DP Standard

The DP standard is the bus protocol of the ET 200 distributed I/O system based on IEC 61784-1:2002 Ed1 CP 3/1.

DPV1

DPV1 designates the functional expansion of the acyclic utilities (e.g. with new alarms) of the DP protocol. The DPV1 functionality is integrated in the IEC 61784-1:2002 Ed1 CP 3/1 standard.

Error Handling via OB

If the operating system detects a specific error (e.g. an access error in the case of *STEP 7*), it calls the organization block (error OB) provided for this event that specifies the subsequent behavior of the CPU.

Error message

An error message is one of the possible responses of the operating system to a runtime error. The other possible responses are: → error response in the user program, STOP mode of the CPU.

Error Response

Response to a → runtime error. The operating system can respond in the following ways: conversion of the programmable controller to STOP mode, call of an organization block in which the user can program a response, or display of the error.

ET 200

The ET 200 distributed I/O system with the PROFIBUS-DP protocol is a bus for connecting distributed I/O devices to a CPU or an appropriate DP master. ET 200 is characterized by a rapid response time, since only a small amount of data (bytes) is transmitted.

ET 200 is based on IEC 61784-1:2002 Ed1 CP 3/1.

The ET 200 works on the master/slave principle. Examples of DP masters are the IM 308-C master interface module and the CPU 315-2 DP.

DP slaves can be the distributed I/O devices ET 200S, ET 200B, ET 200C, ET 200M, ET 200X, ET 200U, ET 200L or DP slaves from Siemens or other vendors.

FC → Function

FORCE

During commissioning, for example, the “Force” function allows certain outputs to be set to “ON” for any length of time, even if the logic operations of the user program are not fulfilled (e.g. because inputs are not wired).

FREEZE

FREEZE is a control command of the DP master to a group of DP slaves.

After the FREEZE control command is received, the DP slave freezes the current state of the **inputs** and transmits it cyclically to the DP master.

After each new FREEZE control command, the DP slave once again freezes the state of the **inputs**.

The input data is transmitted cyclically again from the DP slave to the DP master only after the DP master has sent the UNFREEZE control command.

Function

A function (FC) is, according to IEC 61131-3, a code block without statistical data. A function allows the transmission of parameters in the user program. Thus, functions are ideal for programming frequently recurring complex functions, such as calculations.

Intelligent DP Slave

The defining feature of an intelligent DP slave is that input/output data is not made available to the DP master by a real input/output of the DP slave directly, but by a preprocessing CPU (in this case the IM 151-7 CPU interface module).

Interrupt

The operating system of the CPU recognizes 10 different priority classes that control the processing of the user program. These runtime levels include interrupts, e.g. diagnostic interrupts. When an interrupt is triggered, the operating system automatically calls an assigned organization block in which the user can program the desired response (for example in an FB).

Interrupt, diagnostic → Diagnostic interrupt

Interrupt, process → Process interrupt

Load Memory

The load memory is part of the CPU. It contains objects generated by the programming device. It is implemented either as a plug-in memory card/micro memory card or a permanently integrated memory.

Marker

Markers are components of the system memory of the CPU for storing intermediate results. They can be access on the basis of bits, bytes, words or double words.

Masse

Ground is the total of all interconnected inactive components of a device that are not able to carry a dangerous contact voltage even in the even of a malfunction.

Master

When they are in possession of the token, masters can send data to and request data from other nodes (= active node). An example of a DP master is the CPU 315-2 DP.

Master System

All DP slaves that are assigned to a DP master for either reading or writing plus the DP master itself make up the master system.

MMC

Micro Memory Card Memory module for SIMATIC systems. Can be used as a load memory and portable data carrier.

MPI

The multipoint interface (MPI) is the programming device interface of the SIMATIC S7.

Nesting Depth

One block can be called from within another using block calls. The nesting depth is the number of code blocks called at the same time.

Node

A device that can send, receive or amplify data via the bus, such as a DP master, DP slave, RS 485 repeater or active star coupler.

OB → Organization Block

OB priority

The operating system of the CPU distinguishes between various priority classes, such as cyclic program scanning and process interrupt-driven program scanning. Each priority class is assigned → organization blocks (OB) in which the S7 user can program a response. The OBs have, by default, differing priorities that determine the order in which they are processed if they occur simultaneously or if they interrupt one another.

Operating Mode

The SIMATIC S7 programmable controllers can detect the following operating modes: STOP, → START, RUN.

Operating system of the CPU

The operating system of the CPU organizes all functions and procedures of the CPU that are not linked to a specific control task.

Organization block

Organization blocks (OBs) form the interface between the operating system of the CPU and the user program. The sequence in which the user program is processed is defined in the organization blocks.

Parameter

1. Variable of a *STEP 7* code block

2. Variable to set the behavior of a module (one or more per module). Each module is delivered with a suitable default setting, which can be changed by configuring the parameters in *STEP 7*.

PD → Programming device

PLC → Programmable logic dontroller

Priority class

The operating system of an S7-CPU provides up to 26 priority classes (or “runtime levels“) to which various organization blocks are assigned. The priority classes determine which OBs will interrupt other OBs. If one priority class holds several OBs, the OBs do not interrupt each other but are processed sequentially.

Process Image

The process image is part of the → system memory of the CPU. The signal states of the inputs are written into the process input image at the start of the cyclic program. At the end of the cyclic program, the signal states in the process output image are transferred to the outputs.

Process Interrupt

The process image is part of the → system memory of the CPU. The signal states of the inputs are written into the process input image at the start of the cyclic program. At the end of the cyclic program, the signal states in the process output image are transferred to the outputs.

PROFIBUS

Process **F**ield **Bus** is a German process and field bus standard, defined in IEC 61784-1:2002 Ed1 CP 3/1. It specifies functional, electrical and mechanical characteristics of a bit serial field bus system.

PROFIBUS is available with the protocols DP (the German abbreviation for distributed I/O), FMS (= field bus message specification), PA (= process automation) or TF (= technology (process-related) functions).

PROFIBUS Address

Each bus node must receive a PROFIBUS address to identify it uniquely on the PROFIBUS bus system.

The PC/PD has the PROFIBUS address “0”.

The PROFIBUS addresses 1 to 125 are permissible for the ET 200S distributed I/O system.

Programmable logic controller

A programmable logic controller (PLC) is an electronic controller whose function is stored in the control unit as a program. The configuration and wiring of the device therefore is not determined by the function of the controller. The programmable logic controller is structured like a computer; it consists of a → CPU (central module) with a memory, input/output modules and an internal bus system. The I/O and the programming language are designed to meet the requirements of the control technology.

Programming device

Programming devices are basically personal computers that are suitable for the industrial environment, compact and transportable. They are equipped with special hardware and software for SIMATIC programmable controllers.

Publisher

A sender in direct data communication *See* Direct Communication

Restart

When a CPU is started up (e.g. by switching the mode selector from STOP to RUN or by switching the power on), the organization block OB 100 (complete restart) is executed before cyclic program scanning (OB 1) commences. In the event of a restart, the process image of the inputs is read in and the STEP 7 user program is processed beginning with the first command in OB 1.

Runtime Error

Error which occurs during processing of the user program on the programmable controller (i.e. not in the process).

Scan Cycle Checkpoint

The point during CPU program scanning at which the process image is updated, for example.

SFC → System Function

Slave

A slave may only exchange data with a → master if prompted by the master to do so. By slaves we mean, for example, all DP slaves such as ET 200S, ET 200B, ET 200X, ET 200M, etc.

Stand-alone operation

The device is operated on a stand-alone basis without data exchange to a superordinate master and without direct communication with other DP slaves. All the modules power up using default parameters and with the maximum configuration (32 slots, 64 bytes consistently).

START

The STARTUP mode is run through when the system goes from the STOP mode to the RUN mode.

Can be triggered by the mode selector or after power on or an operator action on the programming device. In the case of the ET 200S a restart is carried out.

Start Event

Start events are defined events such as errors, times and interrupts. They cause the operating system to start an associated organization block (if programmed accordingly by the user). Start events are displayed in the header information of the associated OB. The user can respond to start events in the user program.

STEP 7

Programming language for developing user programs for SIMATIC S7 PLCs.

Subscriber

A recipient in direct communication *See* Direct Communication

SYNC

SYNC is a control command of the DP master to a group of DP slaves.

By means of the SYNC control command, the DP master causes the DP slave to freeze the current statuses of the **outputs**. For the following frames, the DP slave stores the output data but the states of the outputs remain unchanged.

After each new SYNC control command, the DP slave sets the outputs that it has saved as output data. The outputs are not cyclically updated again until the DP master sends the UNSYNC control command.

System diagnostics

System diagnostics is the detection, evaluation and notification of errors that occur within the automation system. Examples of such errors are program errors or module failures. System errors can be indicated by means of LED displays or in *STEP 7*.

System function

A system function (SFC) is a function integrated in the operating system of the CPU that can be called up in the STEP 7 user program as required.

System memory

The system memory is integrated in the central module and designed as a RAM. The system memory includes the address areas (for example timers, counters, memory markers, etc.) as well as the data areas (e.g. communication buffers) required internally by the operating system.

Times

Times are components of the → system memory of the CPU. The contents of the “imer cells” are updated automatically by the operating system asynchronously to the user program. STEP 7 instructions are used to define the exact function of the timer cells (e.g. on-delay) and initiate their execution (e.g. start).

Token

Access right on bus

Total current

Sum of the currents of all output channels of a digital output module.

Transmission rate

The transmission rate is the data transmission speed and specifies the number of transmitted bits per second (transmission rate = bit rate).

In the case of the ET 200S, transmission rates of 9.6 kbaud to 12 Mbaud are possible.

User memory

The user memory contains the code and data blocks of the user program. The user memory can be integrated in the CPU or can be provided on plug-in memory cards (IM 151-7 CPU) or memory modules. However, the user program is always processed from the → working memory of the CPU.

User program

SIMATIC differentiates between → the operating system of the CPU and user programs. The latter are created in the various programming languages (ladder diagram and instruction list) using the → STEP 7 programming software and are stored in code blocks. Data is stored in data blocks.

Working Memory

The working memory is random-access memory in the → CPU that is accessed by the processor while the user program is executed.

Index

E

F

G

I

L

M

N

O

P

R

S

T

U

W

SIEMENS

Product information for

11.2005

Manual Basic Module BM 147 CPU, Version 05/2003

Manual Interface Module IM 151-7 CPU, Version 11/2003

This product information contains important information about the documentation mentioned above. It is to be regarded as a separate component. Its specifications and information have a higher binding nature than those of other manuals and catalogs in case of discrepancies.

Larger working memory and extended number range for blocks

The working memory has been extended for the Basic Module BM 147 CPU and the Interface Module IM 151-7 CPU. The CPUs can now execute larger user programs.
You can now use the block numbers 0 to 2047 for FB and FC in the user programs.

The total amount of blocks (FBs + FCs + DBs) remain unchanged with max. 1024.

	BM 147-1 (6ES7 147-1AA11-0XB0)	**BM 147-2** (6ES7 147-2AA01-0XB0)	**BM 147-2** (6ES7 147-2AB01-0XB0)	**IM 151-7** (6ES7 151-7AA11-0AB0)
Working memory				
• Size • Expandable	64 KB No	64 KB No	128 KB No	64 KB No
Blocks (FB, FC)				
FB • Quantity • Number range	 Max. 512 FB 0 to FB 2047	 Max. 512 FB 0 to FB 2047	 Max. 512 FB 0 to FB 2047	 Max. 512 FB 0 to FB 2047
FC • Quantity • Number range	 Max. 512 FC 0 to FC 2047	 Max. 512 FC 0 to FC 2047	 Max. 512 FC 0 to FC 2047	 Max. 512 FC 0 to FC 2047

A5E00385826-02

Product information for
A5E00385826-02