SIEMENS

SIMATIC

ET 200S Distributed I/O
IM151-7 CPU Interface Module

Operating Instructions

10/2011
A5E00058783-05

Legal information

Warning notice system

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

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

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

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

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

NOTICE
indicates that an unintended result or situation can occur if the relevant information is not taken into account.

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

Qualified Personnel

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

Proper use of Siemens products

Note the following:

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

Trademarks

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

Disclaimer of Liability

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

Siemens AG
Industry Sector
Postfach 48 48
90026 NÜRNBERG
GERMANY

A5E02049034-05
℗ 09/2011

Preface

Purpose of the operating instructions

These operating instructions are intended to supplement the ET 200S Distributed I/O System (http://support.automation.siemens.com/WW/view/en/1144348) Operating Instructions. It contains a description of all the functions performed by the IM 151-7 CPU interface module. The operating instructions do not include functions that relate generally to ET 200S. These can be found in the *ET 200S Distributed I/O System* Operating Instructions.

The information contained in these operating instructions and the *ET 200S Distributed I/O System* Operating Instructions allows you to commission ET 200S with the IM 151-7 CPU interface module and to run it as an I slave on the PROFIBUS DP or in an MPI network. You will also find information on how the IM 151-7 CPU interface module can be operated together with the DP master module on the PROFIBUS DP.

Basic knowledge required

To understand these operating instructions you should have general experience in the field of automation engineering.

Range of validity of these operating instructions

These Operating Instructions are valid for

- the IM 151-7 CPU interface module (order number 6ES7151-7AA21-0AB0)
- the DP master module (order number 6ES7138-4HA00-0AB0)
- the components of the ET 200S distributed I/O system specified in the *ET 200S Distributed I/O System* Operating Instructions.

Note

A description of the special features of the interface module IM151-7F CPU (6ES7151-7FA21-0AB0) can be found in the product information on the Internet (http://support.automation.siemens.com/WW/view/en/11669702/133300).

These operating instructions contain a description of the components that was valid at the time of publication. We reserve the right to issue a Product Information which contains up-to-date information about new components and new versions of components.

Changes since the previous version

Compared to the previous version of this operating instructions *ET 200S, IM151-7 CPU Interface Module*, Edition 11/2003, A5E00058783-04, this edition is a new revision with a completely updated structure. The major changes are:

- Increased work memory
- Increased performance due to shorter instruction processing times
- Number of blocks that can be monitored using the status block increased from 1 to 2
- Effective from *STEP 7* V5.5, increase in the status information that can be monitored using the status block
- Number of breakpoints increased from 2 to 4
- Local data stack size increased (32 KB per execution level/2 KB per block)
- Configurable process image
- Expansion of the block number range
- DB sizes: Max. 64 KB
- Time-delay interrupts: uniform OB 21 / OB 22
- Cyclic interrupts: uniform OB 32 - OB 35
- Increase of GD circles for global data communication
- Number of displayed diagnostic buffer entries in CPU RUN mode is configurable
- Extension of the diagnostic buffer entries in the event of problems on the local I/O bus of the IM151-7 CPU
- Reading service data
- Extension of SFC 12 with 2 new modes to trigger the OB 86 during enabling/disabling of PROFIBUS slaves
- Supports the status byte for power modules
- Encryption of blocks using S7 Block Privacy
- Copying of 512 bytes with SFC 81
- Increase of number of block-related messages (Alarm_S) to 300

Guide

The operating instructions contain the following guides which provide quick access to the specific information you need:

- At the beginning of the documentation you will find a comprehensive table of contents.
- Important terms are explained in the glossary.
- Navigate to the most important topics in our documents using the index.

Recycling and disposal

The IM 151-7 CPU interface module is recyclable due to its non-toxic materials. For environmentally compliant recycling and disposal of your electronic waste, please contact a company certified for the disposal of electronic waste.

Further support

If you have any questions relating to the products described in these operating instructions, and do not find the answers in this document, please contact your Siemens partner at our local offices.

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

A guide to the technical documentation for the various SIMATIC products and systems is available in the Internet (http://www.siemens.com/automation/simatic/portal).

The online catalog and ordering systems are available on the Internet (http://www.siemens.com/automation/mall).

Training center

We offer courses to help you get started with the ET 200S and the SIMATIC S7 automation system. Please contact your regional training center or the central training center in D -90327, Nuremberg, Germany.

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

Service & Support on the Internet

In addition to our documentation, we offer a comprehensive knowledge base on the Internet (http://www.siemens.com/automation/service&support).

There you will find:

- Our Newsletter, which constantly provides you with the latest information about your products.
- The right documentation for you using our Service & Support search engine.
- The bulletin board, a worldwide knowledge exchange for users and experts.
- Your local contact partner for Industry Automation products in our Contact Partners database.
- Information about on-site service, repairs, spare parts, and much more is available under "Repairs, spare parts, and consulting".

Table of contents

1 Description

1.1 Function of the IM 151-7 CPU interface module

The IM 151-7 CPU interface module is a component of the ET 200S distributed I/O system with degree of protection IP20. The IM 151-7 CPU interface module is an "intelligent preprocessor" (I slave). It enables you to decentralize control tasks.

Therefore, an ET 200S with IM 151-7 CPU can exercise full and, if necessary, independent control over a process-related functional unit and can be used as standalone CPU. An IM151-7 CPU can:

- be operated with MPI interface
- either be a DP slave or together with the optional DP master module, a DP master at the PROFIBUS DP.

The use of the IM 151-7 CPU interface module leads to further modularization and standardization of process-related functional units and simple, clear machine concepts.

1.2 Properties of the IM 151-7 CPU interface module

Properties of the IM 151-7 CPU interface module

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

- The interface module has PLC functionality (integrated CPU component with 128 KB work memory).
- The interface module can only be used with the load memory inserted (SIMATIC Micro Memory Card).
- The interface module can be enhanced with up to 63 I/O modules from the ET 200S range.
- The maximum bus length is 2 m.
- Down times are minimized thanks to the integrated diagnostics.
- It is possible to update the firmware via SIMATIC Micro Memory Card or online via the network.
- 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 jobs (FRCE)
  - Operating mode of the IM 151-7 CPU interface module (RUN and STOP)
- Connection to PROFIBUS DP or MPI using a combined MPI / DP interface (RS 485)
- The IM 151-7 CPU interface module can be expanded by **one** DP master module. This also lends it the functionality of a DP master.

Integration of the IM 151-7 CPU interface module in ET 200S

The IM 151-7 CPU interface module is integrated in ET 200S just like any other module, which means the same configuration concept, installation and expansion capability. Information on this can be found in the ET 200S Distributed I/O System (http://support.automation.siemens.com/WW/view/en/1144348) Operating Instructions.

How do I configure and program the ET 200S with IM 151-7 CPU?

You need the configuration software *STEP 7* as of V5.2 + SP1 + HSP219 or V5.5 + SP1 to configure the ET 200S with IM151-7 CPU (configuration and parameter assignment) and to program the IM151-7 CPU interface module.

If you want to operate IO-Link modules with the IM151-7 CPU interface module, you need *STEP 7* as of V5.5 + SP1 or

- *STEP 7* as of V5.4 + SP5 + Hotfix 8 + HSP180 for IO-Link electronic modules + HSP219 for the IM151-7 CPU interface module
- *STEP 7* as of V5.4 + SP5 + Hotfix 8 + HSP181 for SIRIUS electronic modules + HSP219 for the IM151-7 CPU interface module

The procedure for configuration of the ET 200S with IM151-7 CPU is described in section Commissioning (Page 85) of these operating instructions. In the *S7-300 CPUs and ET 200 CPUs Instruction List* you will find the *STEP 7* instruction set for programming the IM151-7 CPU interface module. The instruction list is available as a download from the Internet (http://support.automation.siemens.com/WW/view/en/31977679).

Constraints on using motor starters and ET 200S modules

With central use in an ET 200S with IM 151-7 CPU the following motor starters and ET 200S modules can cause disturbing responses. The product versions specified of these motor starters and ET 200S modules should **not** be used in an ET 200S with IM 151-7 CPU.

Table 1- 1 Constraints on using motor starters and ET 200S modules

Motor starter / module	Order number	Up to and including product version
DS1e-x direct-on-line starter; HF	3RK1301-0❑B10-❑AA2	E06
RS1e-x reversing starters; HF		
F-DS1e-x fail-safe direct starters; HF	3RK1301-0❑B13-❑AA2	E06
F-RS1e-x fail-safe reversing starters; HF		
DS1e-x direct-on-line starter; HF	3RK1301-0❑B❑0-❑AA3	E03
RS1e-x reversing starters; HF		
DSS1e-x direct soft starters; HF		
DS1e-x direct-on-line starter; HF	3RK1301-0❑B❑❑-❑AA4	E02
RS1e-x reversing starters; HF		
DSS1e-x direct soft starters; HF		
F-DS1e-x fail-safe direct starters; HF		
F-RS1e-x fail-safe reversing starters; HF		
2AI I 2WIRE HS analog electronic module	6ES7134-4GB52-0AB0	E03
2 AI I 4WIRE HS analog electronic module	6ES7134-4GB62-0AB0	E01
Analog electronic module 2AI U HS	6ES7134-4FB52-0AB0	E01
2AO I HS analog electronic module	6ES7135-4GB52-0AB0	E01
2AO U HS analog electronic module	6ES7135-4FB52-0AB0	E03

1.3 Properties of the DP master module

Together with the DP master module you can operate the IM 151-7 CPU interface module as a DP master.

Note

The IM 151-7 CPU interface module can be expanded by no more than **one** DP master module.

Properties of the DP master module

The DP master module has the following special features:

- The PROFIBUS DP address is saved alongside the HW Config configuration on the SIMATIC Micro Memory Card in the IM 151-7 CPU interface module.
- There is 1 LED on the front of the DP master module to indicate bus faults on the PROFIBUS DP (BF).
- Connection to PROFIBUS DP via the DP interface (RS 485) on the DP master module

Integration of the DP master module in ET 200S

The DP master module is connected to the IM 151-7 CPU from the right and hence integrated in the ET 200S.

How do I configure and program the ET 200S with IM 151-7 CPU and DP master module?

To configure an ET 200S with IM 151-7 CPU and DP master module (configuration and parameter assignment) and to program the IM 151-7 CPU interface module you will need the *STEP 7* configuration software as of V5.2 + SP1 + HSP219 or V5.5 + SP1.
The procedure for configuring the ET 200S with IM 151-7 CPU and DP master module is described in the section Commissioning (Page 85) of these operating instructions.

1.4 Example configurations

Example configuration of an ET 200S with IM 151-7 CPU

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

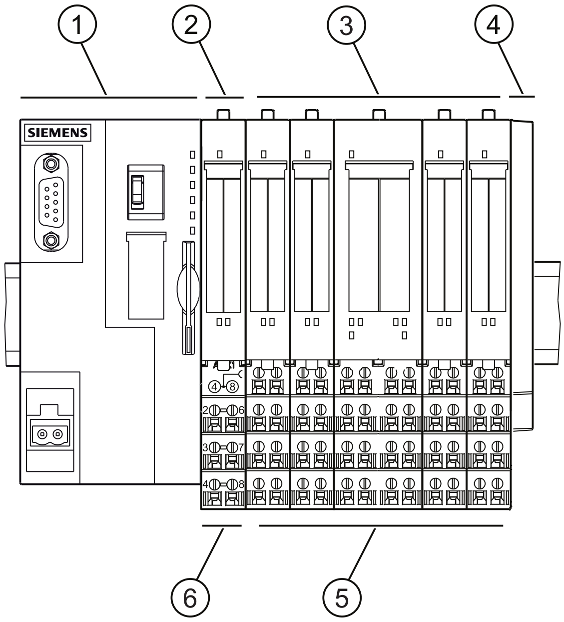


Figure 1-1 View of the ET 200S distributed I/O system with IM 151-7 CPU

① IM151-7 CPU interface module

② PM-E power module for electronic modules

③ Electronic modules

④ Terminating module

⑤ TM-E terminal modules for electronic modules

⑥ TM-P terminal modules for PM-E power modules

Example configuration of an ET 200S with IM151-7 CPU and DP master module

The figure below shows an example configuration of an ET 200S with IM151-7 CPU and DP master module.

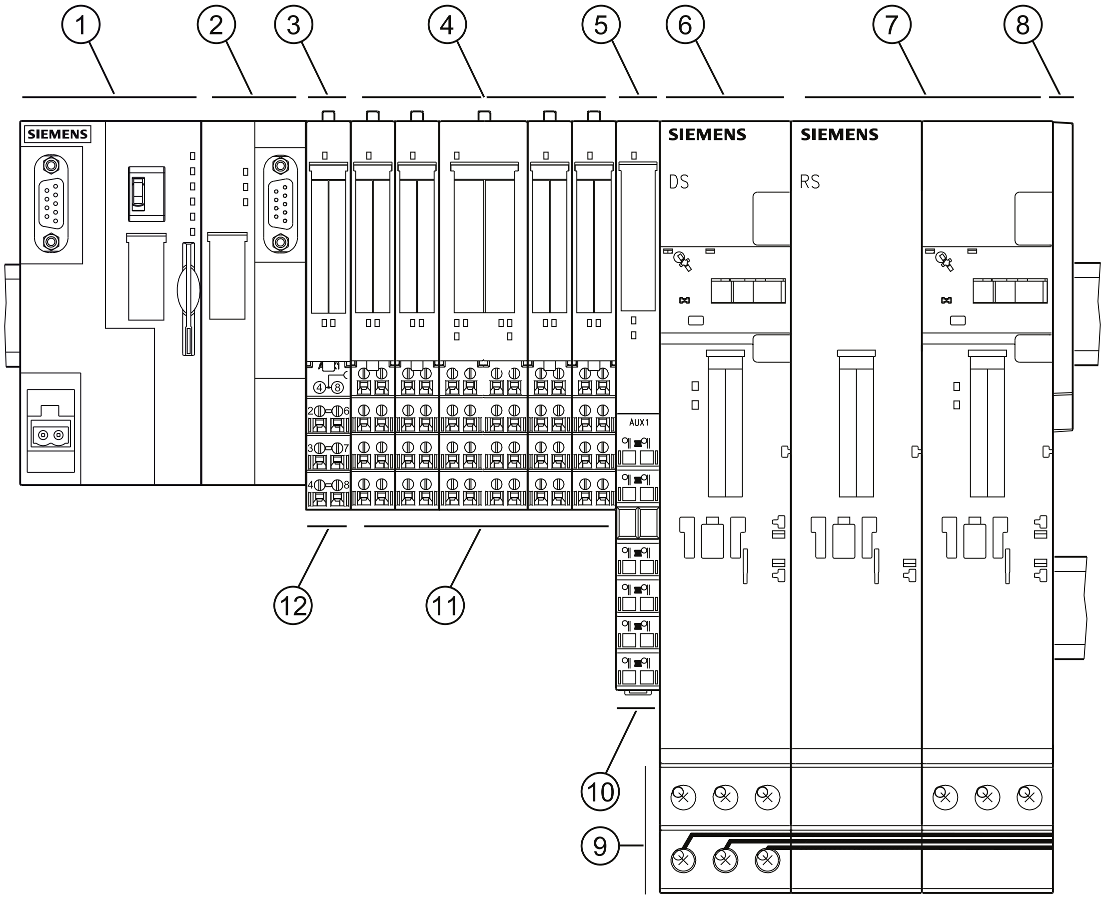
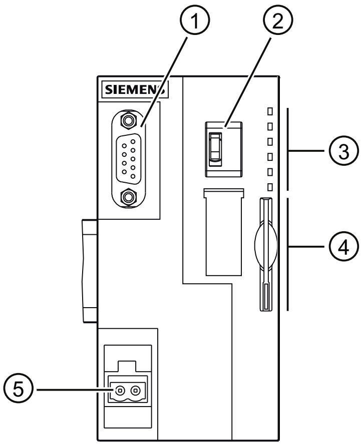


Figure 1-2 View of the ET 200S distributed I/O system with IM151-7 CPU and DP master module

① IM151-7 CPU interface module
② DP master module
③ PM-E power module for electronic modules
④ Electronic modules
⑤ PM-D power module for motor starters
⑥ Direct starters
⑦ Reversing starter
⑧ Terminating module
⑨ Power bus
⑩ TM-P terminal module for PM-D power modules
⑪ TM-E terminal modules for electronic modules
⑫ TM-P terminal modules for PM-E power modules

Operating and display elements 2

2.1 Operating and display elements of the IM151-7 CPU interface module



Number	Designation
①	Combined MPI / DP interface (RS 485) for connection to PROFIBUS DP or MPI
②	Mode selector switch
③	Status and error displays of the IM151-7P CPU interface module
④	Slot for the SIMATIC Micro Memory Card
⑤	Connection for supply voltage

Slot for the SIMATIC Micro Memory Card

Memory module is a SIMATIC Micro Memory Card. You can use MMCs as load memory and as portable storage media. The slot for the SIMATIC Micro Memory Card can be accessed from the front of the interface module. Detailed information on inserting the SIMATIC Micro Memory Card is available in the section Inserting/Replacing a SIMATIC Micro Memory Card (Page 90).

Note

The IM151-7 CPU interface module does not have an integrated load memory, so you will need to connect a SIMATIC Micro Memory Card to the IM 151-7 interface module in order to use it.

Mode selector switch

You can use the mode selector switch to set the current operating mode of the IM151-7 CPU.

Table 2- 1 Mode selector switch settings

Position	Meaning	Description
RUN	RUN mode	The IM151-7 CPU interface module processes the user program.
STOP	STOP mode	The IM151-7 CPU interface module does not process the user program.
MRES	Memory reset	Mode selector switch setting for • Memory reset of the IM151-7 CPU interface module • Backing up the firmware to the SIMATIC Micro Memory Card • Resetting to the as-supplied state A memory reset using the mode selector requires a number of steps to be carried out in a set order.

Reference

- Operating modes of the IM151-7 CPU interface module: *STEP 7 Online Help*
- Information on memory reset of the IM 151-7 CPU interface module: Section Memory reset of the IM151-7 CPU interface module using the mode selector switch (Page 93)
- Backing up the firmware to the SIMATIC Micro Memory Card: Section Backing up firmware on a SIMATIC Micro Memory Card (Page 125)
- Resetting to factory settings: Section Resetting to the as-delivered state (Page 98)
- Evaluation of the LEDs for errors or diagnostics: Section Diagnostics using status and error LEDs (Page 150)

2.2 Status and error displays of the IM151-7 CPU interface module

Status and error LEDs

Table 2- 2 Status and error displays of the IM151-7P CPU interface module

LED designation	Color	Meaning
SF	Red	Group fault for hardware or software error
BF	Red	Bus fault at PROFIBUS DP
ON	Green	Supply voltage for the IM151-7 CPU
FRCE	Yellow	LED is lit: Active force job LED flashes at 2 Hz: Node flash test function.
RUN	Green	IM151-7 CPU in RUN The LED flashes during STARTUP at a rate of 2 Hz, and in HOLD state at 0.5 Hz.
STOP	Yellow	IM151-7 CPU in STOP or in HOLD or STARTUP mode The LED flashes at 0.5 Hz when the CPU requests a memory reset, and during the reset at 2 Hz.

Reference

- Operating modes of the IM151-7 CPU interface module: *STEP 7 Online Help*
- Information on memory reset of the IM 151-7 CPU interface module: Section Memory reset of the IM151-7 CPU interface module using the mode selector switch (Page 93)
- Evaluation of the LEDs for errors or diagnostics: Section Diagnostics using status and error LEDs (Page 150)

2.3 Display elements of the DP master module

Display elements

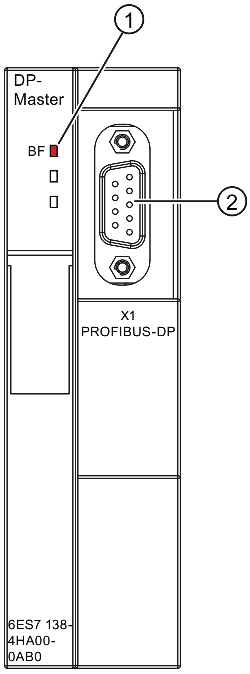


Figure 2-1 Display elements of the DP master module

The figure shows ...	the following elements of the DP master module
①	Status and error displays
②	9-pin sub D socket for PROFIBUS DP

3 Communication

3.1 Interfaces

3.1.1 Multi-Point Interface (MPI)

Availability

The IM151-7 CPU interface module has a combined MPI / DP interface (X1) with DP slave functionality. This RS 485 interface is configured as an MPI interface when the unit is first supplied.

Properties

The MPI (Multi-Point Interface) is the interface of the IM151-7 CPU interface module for PG/OP connections or for communication on an MPI subnet.

The typical (default) baud rate is 187.5 Kbps. You can also set 19.2 Kbps for communication with a S7-200. Baud rates up to 12 Mbps are possible.

The IM151-7 CPU interface module automatically broadcasts its bus configuration via the MPI interface (the transmission rate, for example). A programming device, for example, can thus receive the correct parameters and automatically connect to a MPI subnet.

Devices capable of MPI communication

- Programming device / PC
- OP/TP
- S7-300 / S7-400 with MPI interface
- S7-200 (only at 19.2 Kbps)

NOTICE
You may only connect programming devices to an MPI subnet which is in RUN mode. Do not connect nodes (for example, OP, TP) to the MPI subnet while the system is running. Otherwise, transferred data might be corrupted as a result of interference, or global data packages may be lost.

Time synchronization

Time synchronization is possible if the MPI / DP interface is programmed as an MPI interface on the IM151-7 CPU interface module. For information on time synchronization via MPI, please refer to the section Time synchronization (Page 44).

3.1.2 PROFIBUS DP

Availability

The IM151-7 CPU interface module has a combined MPI / DP interface (X1) with **DP slave functionality**. This RS 485 interface is configured as an MPI interface when the unit is first supplied. If you want to use the DP interface, you have to reconfigure it in *STEP 7* as DP interface.

Together with the optional DP master module, the IM 151-7 CPU interface module has an RS 485 interface (X2) with **DP master functionality**.

Properties of the PROFIBUS DP interfaces

PROFIBUS DP interface at IM151-7 CPU

The PROFIBUS DP interface (**X1**) with **DP slave functionality** is mainly used to connect the IM151-7 CPU interface module to a PROFIBUS DP subnet (use of CPU as I slave).

The PROFIBUS DP interface offers a transmission rate of up to 12 Mbps.

You can configure the DP slave interface as active or passive by using the "Test, Commissioning, Routing" checkbox in the properties of the DP interface in *STEP 7*.

- Active DP slave interface ("Test, Commissioning, Routing" checkbox is activated):
  - The PROFIBUS DP interface broadcasts its bus configuration (the transmission rate, for example). A programming device, for example, can thus receive the correct parameters and automatically connect to a PROFIBUS subnet. In your configuration you can specify to disable bus parameter broadcasting.
  - Routing/data set routing in other subnets that can be accessed by an additionally connected DP master module is possible.
  - PG/OP functions (e. g. for testing and commissioning) are possible without restrictions.
  - Do not operate a DP slave with active DP interface in an isochronous subnet.
- Passive DP slave interface ("Test, Commissioning, Routing" checkbox is deactivated):
  - The configured transmission rate is ignored and set automatically to the transmission rate of the master (automatic transmission rate search)
  - Routing/data set routing is no longer possible via a passive DP slave interface.
  - PG/OP functions are possible but with less performance than with active interface.
  - If you want to operate the CPU as I slave in a subnet in which isochronous DP slaves are used, the DP slave interface of the I slave CPU must be operated passively (the CPU as I slave does not support isochronous slave operation itself)

DP master interface on the DP master module

The DP master interface on the DP master module (X2) is mainly used to connect distributed I/O (slaves). PROFIBUS DP allows you to create large subnets.

The DP master interface offers a transmission rate of up to 12 Mbps.

You can configure the DP master interface as DP master or to be inactive.

- The interface needs a configuration as DP master.
  - Slaves can be operated with loaded configuration; PG/OP functions and routing are possible.
  - The CPU sends its set bus parameters (the baud rate, for example) via the PROFIBUS DP interface if master mode is set. This functionality automatically provides the correct parameters for online operation of a programming device, for example. In your configuration you can specify to disable bus parameter broadcasting.
- The interface is always inactive without configuration.

Devices capable of PROFIBUS DP communication

Devices that can be connected via PROFIBUS DP:

- Programming device / PC
- OP/TP
- DP slaves
- DP master
- Actuators/Sensors
- S7-300/S7-400 with PROFIBUS DP interface

Time synchronization using PROFIBUS

Time synchronization is possible if the MPI / DP interface is programmed as a DP interface on the IM151-7 CPU interface module and via the DP interface on the DP master module. For information on time synchronization via PROFIBUS DP, please refer to the section Time synchronization (Page 44).

Reference

Additional information on PROFIBUS: "PROFIBUS (http://www.profibus.com)"

3.2 Configuring MPI and PROFIBUS subnets

3.2.1 Basic information relating to MPI and PROFIBUS subnets

Convention: device = node

All devices you interconnect on the MPI or PROFIBUS network are referred to as nodes.

Segment

A segment is a bus line between two terminating resistors. A segment may contain up to 32 nodes. It is also limited with respect to the permitted line length, which is determined by the transmission rate.

Baud rate

Maximum transmission rates:

- MPI:
  - 12 Mbps
  - Default: 187,5 Kbps
- PROFIBUS DP: 12 Mbps

Number of nodes

Maximum number of nodes per subnet.

Table 3- 1 Subnet nodes

Parameters	MPI	PROFIBUS DP
Number	127	126 [1]
Addresses	0 to 126	0 to 125
Comment	Default: 32 addresses Reserved addresses: • Address 0 for PG • Address 1 for OP	of those: • 1 master (reserved) • 1 PG connection (address 0 reserved) • 124 [1] slaves or other masters
[1] Note that the IM151-7 CPU interface module as DP master (with DP master module) can only operate up to 32 slaves.		

MPI/PROFIBUS DP addresses

You need to assign an address to all nodes in order to enable intercommunication:

- On the MPI network: an "MPI address"
- On the PROFIBUS DP network: "a PROFIBUS DP address"

You can use the PG to set the MPI/PROFIBUS addresses for each one of the nodes (some of the PROFIBUS DP slaves are equipped with a selector switch for this purpose).

Default MPI/PROFIBUS DP addresses

The table below shows you the default setting of the MPI/PROFIBUS DP addresses, and the factory setting of the highest MPI/PROFIBUS DP addresses for the nodes.

Table 3- 2 MPI/PROFIBUS DP addresses

Node (device)	Default MPI/ PROFIBUS DP address	Default highest MPI address	Default highest PROFIBUS DP address
PG	0	32	126
OP	1	32	126
CPU	2	32	126

Rules: Assignment of MPI/PROFIBUS DP addresses

Note the following rules before assigning MPI/PROFIBUS addresses:

- All MPI/PROFIBUS subnet addresses must be unique.
- The highest MPI/PROFIBUS address must be greater than or equal to the actual MPI/ PROFIBUS address and must be identical for each node. (Exception: Connecting the PG to several nodes).

Recommendation for MPI addresses

Reserve MPI address "0" for a service PG, or "1" for a service OP, for temporary connections of these devices to the subnet. You should therefore assign different MPI addresses to PGs/OPs operating on the MPI subnet.

Recommended MPI address of the CPU for replacement or service operations:

Reserve MPI address "2" for the CPU. This prevents duplication of MPI addresses after you connect a CPU with default settings to the MPI subnet (for example, when replacing a CPU). You should assign an MPI address greater than "2" to CPUs in the MPI subnet.

Recommendation for PROFIBUS addresses

Reserve PROFIBUS address "0" for a service PG that you can subsequently connect briefly to the PROFIBUS subnet as required. You should therefore assign unique PROFIBUS addresses to PGs integrated in the PROFIBUS subnet.

PROFIBUS DP: Electrical cables or fiber-optic cables?

Use fiber optic cables on a field bus with greater length, rather than copper conductors, in order to be independent on the transmission rate, and to exclude external interference.

3.2.2 Network components of MPI/DP and cable lengths

MPI subnet segment

You can install cables with a length of up to 50 m in an MPI subnet segment. This length of 50 m is the distance between the first and the last node of the segment.

Table 3- 3 Permissible cable length of a segment on the MPI subnet

Baud rate	Maximum cable length of a segment
19,2 Kbps	1000 m
187,5 Kbps	
1,5 Mbps	200 m
3,0 Mbps	100 m
6,0 Mbps	
12,0 Mbps	

Segment on the PROFIBUS subnet

The maximum cable length of a segment on the PROFIBUS subnet is determined by the set transmission rate.

Table 3- 4 Permissible cable length of a segment on the PROFIBUS subnet

Baud rate	Maximum cable length of a segment
9,6 Kbps to 187.5 Kbps	1000 m
500 Kbps	400 m
1,5 Mbps	200 m
3 Mbps to 12 Mbps	100 m

Longer cable lengths via RS485 repeater/RS485 diagnostic repeater

You need to install RS485 repeaters for segments requiring cable lengths longer than the allowed length. For additional information about the RS485 Repeater, refer to the RS 485 Repeater Manual (http://support.automation.siemens.com/WW/view/en/48598071).

Stub cables

Make allowances for the maximum stub cable length when you connect bus nodes to a segment by means of stub cables, for example, a PG via standard PG cable.

For transmission rates up to 3 Mbps, you can use a PROFIBUS bus cable with bus connector as stub cable for connecting. For transmission rates of 3 Mbps and higher, use the PG patch cord to connect the PG or PC. You can use multiple PG connecting cables in one bus structure. Other types of stub cables are not permitted.

Length of stub cables

The table below shows the maximum permitted lengths of stub cables per segment:

Table 3- 5 Lengths of stub cables per segment

Baud rate	Max. length of stub cables per segment	Number of nodes with stub cable length of ...	
		1.5 m or 1.6 m	3 m
9,6 Kbps to 93.75 Kbps	96 m	32	32
187,5 Kbps	75 m	32	25
500 Kbps	30 m	20	10
1,5 Mbps	10 m	6	3
3 Mbps to 12 Mbps	[1]	[1]	[1]

[1] To connect PGs or PCs for operation at transmission rates starting at 3 Mbps, use the PG patch cord with order number 6ES7901-4BD00-0XA0. In a bus configuration, you can use multiple PG patch cords with this order number. Other types of stub cables are not permitted.

PG connecting cable

Table 3- 6 PG connecting cable

Type	Order number
PG connecting cable	6ES7901-4BD00-0XA0

PROFIBUS cables

For PROFIBUS DP or MPI networking we offer you the following bus cables for diverse fields of application:

Table 3- 7 Available bus cables

Bus cable	Order number
PROFIBUS cable	6XV1830-0AH10
PROFIBUS cable, halogen-free	6XV1830-0LH10
PROFIBUS underground cable	6XV1830-3FH10
PROFIBUS trailing cable	6XV1830-3BH10
PROFIBUS cable with PUR sheath for environments subject to chemical and mechanical stress	6XV1830-0JH10
PROFIBUS cable with PE sheath for the food and beverages industry	6XV1830-0GH10
PROFIBUS cable for festooning	6XV1830-3GH10

Properties of PROFIBUS cables

The PROFIBUS bus cable is a 2-wire, shielded twisted pair cable with copper conductors. It is used for hardwired transmission in accordance with US Standard EIA RS485.

The table below lists the characteristics of these cables.

Table 3- 8 Properties of PROFIBUS cables

Properties	Values
Wave impedance	approx. 135 Ω to 160 Ω (f = 3 MHz to 20 MHz)
Loop resistance	≤ 115 Ω/km
Effective capacitance	30 nF/km
Attenuation	0.9 dB/100 m (f = 200 kHz)
Permitted conductor cross-sections	0.3 mm^2 to 0.5 mm^2
Permitted cable diameter	8 mm ± 0.5 mm

Installation of bus cables

When you install PROFIBUS bus cables, you must not

- twist,
- stretch
- or compress them.

When wiring indoor bus cables, also maintain the following marginal conditions (d_A = outer cable diameter):

Table 3- 9 Marginal conditions for wiring interior bus cables

Characteristics	Condition
Bending radius (one-off)	≥80 mm (10 x d_A)
Bending radius (multiple times)	≥160 mm (20 x d_A)
Permitted temperature range during installation	-5 °C to +50 °C
Shelf and static operating temperature range	-30 °C to +65 °C

Reference

For information on the use of fiber-optic cables for PROFIBUS bus cables, refer to the SIMATIC NET PROFIBUS (http://support.automation.siemens.com/WW/view/en/35222591) Networks Manual.

RS 485 bus connector

Suitable bus connectors are listed in the section Order numbers of accessories (Page 194).

Field of application

You need bus connectors to connect the PROFIBUS bus cable to an MPI or PROFIBUS DP interface.

You do not require a bus connector for:

- DP slaves with degree of protection IP 65 (ET 200pro, for example)
- RS 485 repeater

RS 485 repeater

Type	Order number
RS 485 repeater	6ES7972-0AA01-0XA0
RS 485 Diagnostic Repeater	6ES7972-0AB01-0XA0

Note

SFC 103 "DP_TOPOL" can be used to initiate identification of the bus topology of a DP master system by way of the interconnected diagnostic repeaters.

Purpose

RS485 repeaters are used to amplify data signals on bus lines and to couple bus segments.

You require an RS 485 Repeater in the following situations:

- more than 32 network nodes
- when interconnecting a grounded with an ungrounded segment
- when exceeding the maximum line length in a segment

Longer cable lengths

If you want to implement cable lengths above those permitted in a segment, you must use RS485 repeaters. The maximum cable lengths between two RS 485 repeaters correspond to the maximum cable length of a segment. Please note that these maximum cable lengths only apply if there is no additional node interconnected between the two RS 485 repeaters. You can connect up to nine RS 485 repeaters in series. Note that you have to include the RS 485 repeater when counting the number of subnet nodes, regardless of whether or not it was assigned its own MPI/PROFIBUS address.

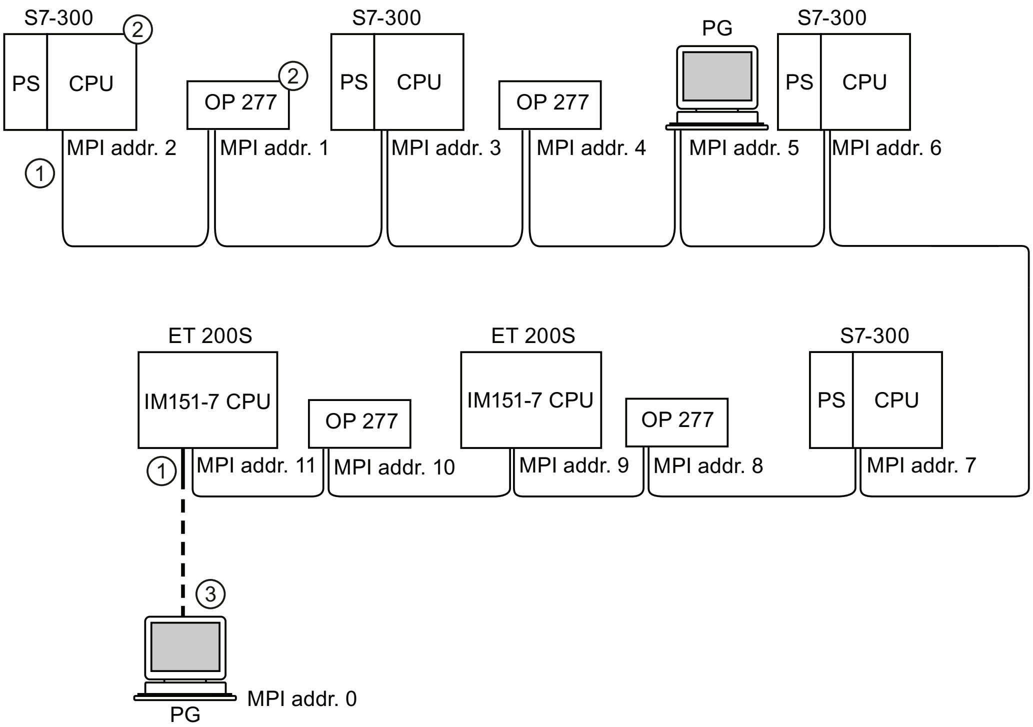
Reference

For additional information about the RS485 Repeater, refer to the RS 485 Repeater Manual (http://support.automation.siemens.com/WW/view/en/48598071).

3.2.3 Examples for MPI and PROFIBUS subnets

Example: Installation of an MPI subnet

The figure below shows you the block diagram of a MPI subnet.

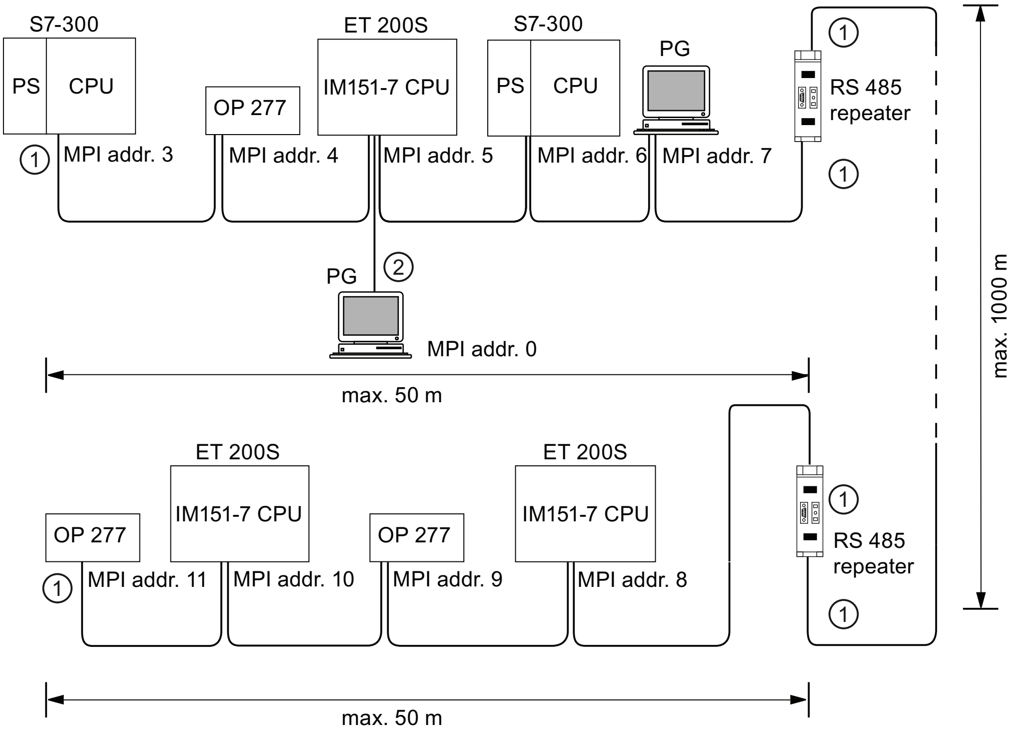


Number	Description
①	Terminating resistor enabled
②	S7-300 and OP 277 have subsequently been connected to the MPI subnet using their default MPI address.
③	Connected via stub cable using the default MPI address only for commissioning/maintenance.

Example: Maximum distances in the MPI subnet

The figure below shows you:

- a possible MPI subnet configuration
- maximum distances possible in an MPI subnet
- the principle of "Line extension" using RS 485 repeaters

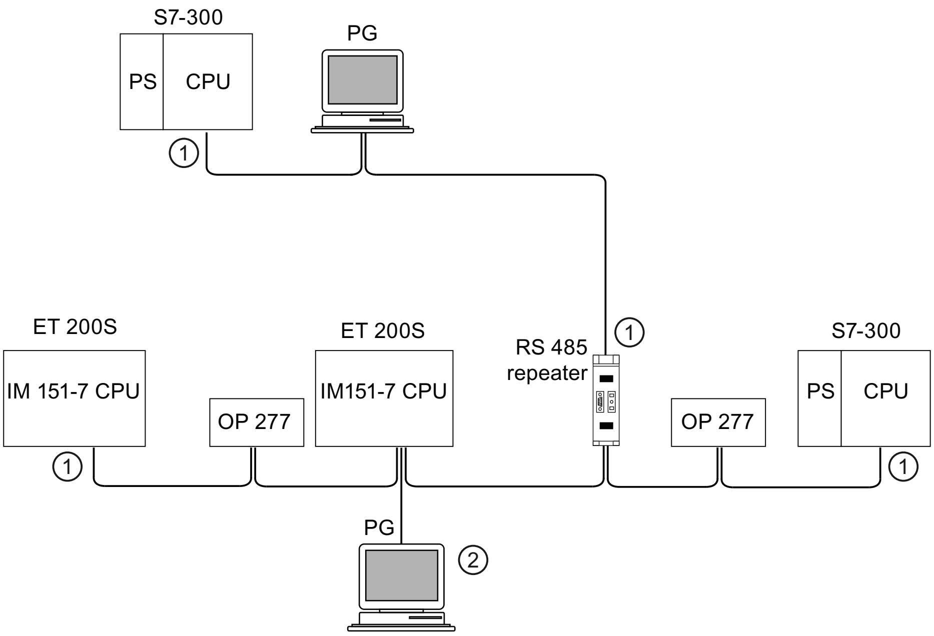


Number	Description
①	Terminating resistor enabled
②	PG connected by means of stub cable for maintenance purposes

Example: Terminating resistor in the MPI subnet

The figure below shows you an example of an MPI subnet and where to enable the terminating resistor.

The figure below illustrates where the terminating resistors must be enabled in an MPI subnet. In this example, the programming device is connected via a stub cable only for the duration of commissioning or maintenance.



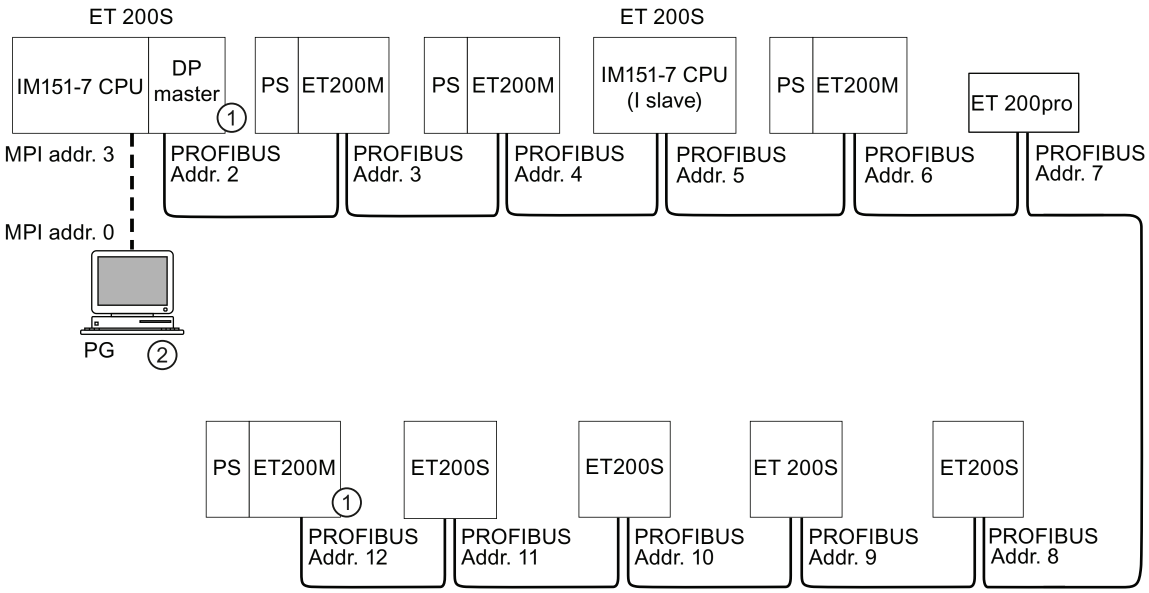
Number	Description
①	Terminating resistor enabled
②	PG connected by means of stub cable for maintenance purposes

WARNING

Disturbance of data traffic might occur on the bus. A bus segment must always be terminated at both ends with the terminating resistor. This, for example, is not the case if the last slave with bus connector is de-energized. Because the bus connector takes its voltage from the station, this terminating resistor is ineffective. Please make sure that power is always supplied to stations on which the terminating resistor is active. Alternatively, the PROFIBUS terminator can also be used as active bus termination.

Example: Installation of a PROFIBUS subnet

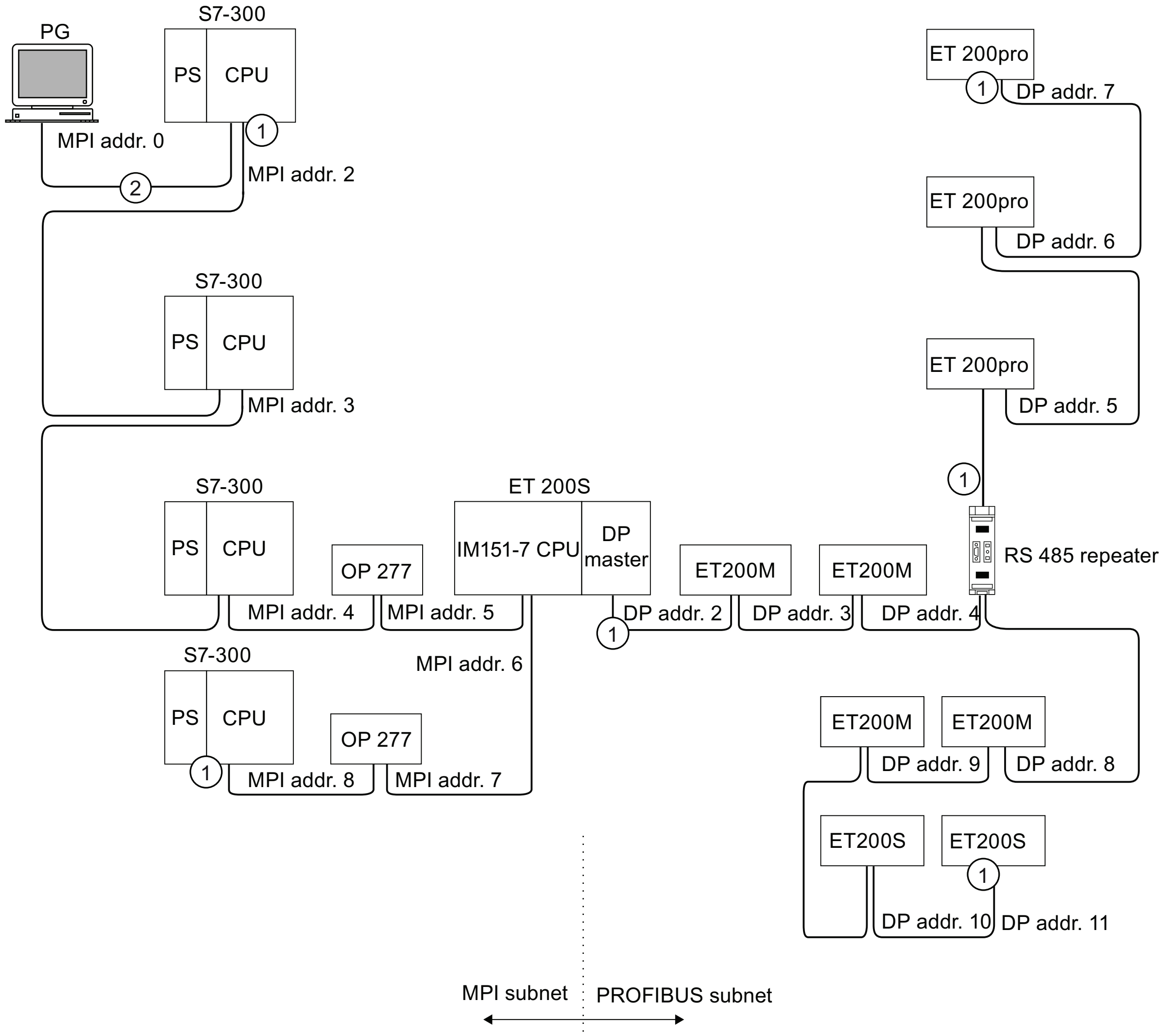
The figure below shows you the basic principles of a PROFIBUS subnet installation.



Number	Description
①	Terminating resistor enabled
②	PG connected by means of stub cable for maintenance purposes

Example: IM151-7 CPU as MPI and PROFIBUS node

The figure below shows you an assembly with the IM151-7 CPU interface module integrated in an MPI subnet and also operated as DP master in a PROFIBUS subnet.



Number	Description
①	Terminating resistor enabled
②	PG connected via a stub cable for maintenance or commissioning purposes

3.3 Communication services

3.3.1 Overview of communication services

Selecting the communication service

You need to decide on a communication service, based on functionality requirements. Your choice of communication service will have no effect on:

- The functionality available
- Whether an S7 connection is required or not
- The time of connecting

The user interfaces can vary considerably (SFC, SFB, etc.) and is also determined by the hardware used (IM151-7 CPU, PC, etc.).

Overview of communication services

The table below provides an overview of the communication services provided by the IM 151-7 CPU interface module.

Table 3- 10 Communication services of the IM151-7 CPU interface module

Communication service	Functionality	Time at which the S7 connection is established ...	via MPI	via DP
Programming device communication	Commissioning, test, diagnostics	From the programming device, starting when the service is used	X	X
OP communication	Control and monitoring	From the OP at Power ON	X	X
S7 basic communication	Data exchange	Programmed to take place via blocks (SFC parameters)	X	I communication (I_PUT / I_GET) [1]
S7 communication	Data exchange in server and client mode: Configuration of communication required	from the active partner at power on.	Only in server mode	Only in server mode
Global data communication	Cyclic data exchange (for example, bit memory)	Does not require an S7 connection	X	–
Routing programming device functions	for example testing, diagnostics on other networks also	From the programming device, starting when the service is used	X[1]	X[1]
Data set routing	for example, parameterization and diagnostics of field devices on the PROFIBUS DP by an engineering system operated on an MPI interface (e.g. PDM)	From the programming device, starting when the service is used	X[1]	X[1]

Communication service	Functionality	Time at which the S7 connection is established ...	via MPI	via DP
PROFIBUS DP	Data exchange between master and slave	Does not require an S7 connection	–	X[2]
Clock synchronization	Broadcast telegrams	Does not require an S7 connection	X	X[3]

[1] with inserted DP master module only

[2] at the integrated MPI/DP interface as DP slave or with inserted DP master module also as DP master

[3] Clock synchronization is only possible as time slave at the integrated MPI/DP interface (DP slave functionality only). with inserted DP master module: Clock synchronization is possible as time slave as well as time master with DP master functionality.

See also

Distribution and availability of S7 connection resources (Page 49)

Connection resources for routing (Page 50)

3.3.2 PG communication

Properties

Programming device communication is used to exchange data between engineering stations (programming device, PC, for example) and SIMATIC modules which are capable of communication. The service is possible via the MPI and PROFIBUS subnets. Transition between subnets is also supported.

Programming device communication provides the functions needed to download / upload programs and configuration data, to run tests and to evaluate diagnostic information. These functions are integrated in the operating system of the IM 151-7 CPU interface module.

An IM 151-7 CPU interface module can maintain several simultaneous online connections to one or multiple programming devices.

3.3.3 OP communication

Properties

OP communication is used to exchange data between operator CPUs (OP, TP, WinCC, for example) and SIMATIC modules which are capable of communication. The service is possible via the MPI and PROFIBUS subnets.

OP communication provides functions you require for monitoring and modifying. These functions are integrated in the operating system of the IM 151-7 CPU interface module.

An IM 151-7 CPU interface module can maintain several simultaneous connections to one or multiple OPs.

3.3.4 S7 basic communication

Properties

Use the S7 basic communication to exchange data between different S7-CPUs (e. g. IM151-7 CPU) on a MPI subnet or with distributed SIMATIC modules that can communicate within an S7 station (e. g. with a FM354 positioning module that is used in an ET 200M slave operated on a DP master module).

The data is exchanged across non-configured S7 connections. The service is available via the MPI subnet or via a DP subnet (only I communication at the inserted DP master module to a distributed function module).

S7 basic communication provides the functions you need to exchange data. These functions are integrated in the operating system of the IM 151-7 CPU interface module. The user can utilize this service by means of "System function" (SFC) user interface.

Reference

Additional information

- to SFCs can be found in the S7300-CPUs and ET 200-CPUs Instruction List (http://support.automation.siemens.com/WW/view/en/31977679).

  A detailed description is available in the *STEP 7 online help* or in the System and standard functions for S7-300/400 (http://support.automation.siemens.com/WW/view/en/1214574) Reference Manual.
- On communication is available in the Communication with SIMATIC (http://support.automation.siemens.com/WW/view/en/25074283) System Manual.

3.3.5 S7 communication

Properties

The IM151-7 CPU interface module can only be server in the S7 communication. The connection is always established by the communication partner. The service is possible via the MPI and PROFIBUS subnets.

The services are handled by the operating system without explicit user interface.

Reference

For additional information on communication, refer to the Communication with SIMATIC (http://support.automation.siemens.com/WW/view/en/25074283) manual.

3.3.6 Global data communication (via MPI only)

Properties

Global data communication is used for cyclic exchange of global data via MPI subnets (for example, I, Q, M) between SIMATIC S7 CPUs (data exchange without acknowledgement). One CPU broadcasts its data to all other CPUs on the MPI subnet. The function is integrated in the operating system of the IM 151-7 CPU interface module.

Reduction ratio

The reduction ratio specifies the cyclic intervals for GD communication. You can set the reduction ratio when you configure global data communication in *STEP 7*. For example, if you set a reduction ratio of 7, global data are transferred only with every 7th cycle. This will reduce the load on the IM151-7 CPU interface module.

Send and receive conditions

Conditions which should be satisfied for GD communication:

- The transmitter of a GD packet must meet the following requirement:
  $\text{Reduction ratio}_{\text{transmitter}}$ x $\text{cycle time}_{\text{transmitter}} \geq 60$ ms
- The receiver of a GD packet must meet the following requirement:
  $\text{Reduction ratio}_{\text{receiver}}$ x $\text{cycle time}_{\text{receiver}}$
  $<$ $\text{reduction factor}_{\text{transmitter}}$ x $\text{cycle time}_{\text{transmitter}}$

Information on cycle time is available in the Appendix Cycle and response times (Page 197).

A GD packet may be lost if these requirements are not met. The reasons being:

- The performance of the "smallest" CPU in the GD circuit
- Asynchronous transmitting / receiving of global data at the stations

When setting in *STEP 7*: "Transmit after each CPU cycle" and the IM151-7 CPU has a short CPU cycle time (< 60 ms), the operating system might overwrite a GD package of the IM151-7 CPU before it is transmitted. The loss of global data is indicated in the status box of a GD circuit, if you set this function in your *STEP 7* configuration.

GD resources of the IM151-7 CPU interface module

Table 3- 11 GD resources of the IM151-7 CPU interface module

Parameters	IM151-7 CPU
Number of GD circuits per CPU	Max. 8
Number of GD packets transmitted per GD circuit	Max. 1
Number of GD packets transmitted by all GD circuits	Max. 8
Number of GD packets received per GD circuit	Max. 1
Number of GD packets received by all GD circuits	Max. 8
Data length per GD packet	Max. 22 bytes
Consistency	Max. 22 bytes
Min. reduction ratio (default)	1 (8)

3.3.7 Routing

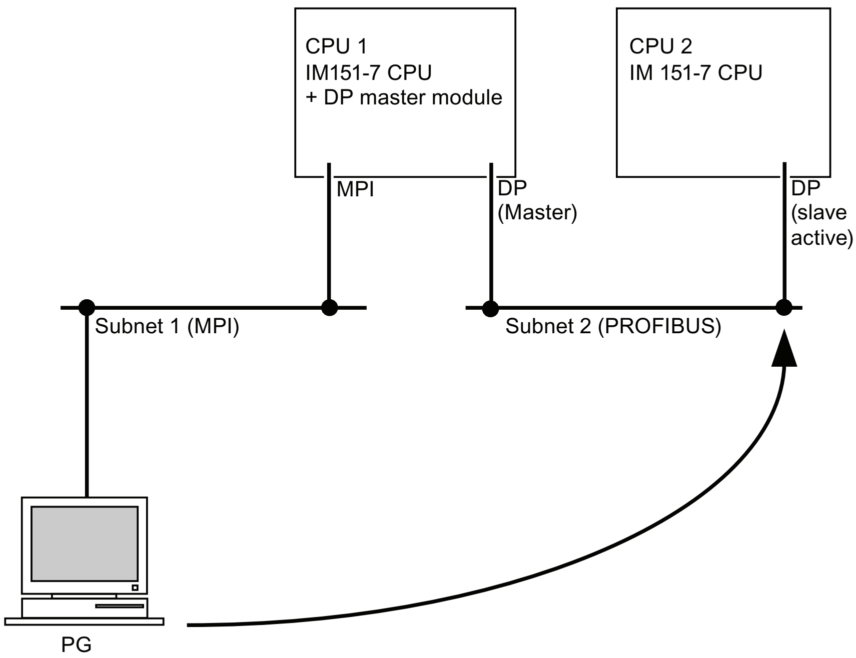
Properties

Using *STEP 7* you can use the PG / PC via your IM151-7 CPU interface module (with DP master module) to reach a CPU in a different subnet, for example, to

- download user programs
- download a hardware configuration or
- perform tests and diagnostics functions.

Routing network nodes: MPI - PROFIBUS

Gateways between subnets are routed in a SIMATIC station that is equipped with interfaces to the respective subnets. The figure below shows the access from MPI to PROFIBUS. The CPU 1 (IM 151-7 CPU with DP master module) is the router between subnet 1 and subnet 2.



Number of connections for routing

For the routing function (only relevant with inserted DP master module) you have up to four connections available with the IM151-7 CPU interface module.

Requirements

- The station modules are "capable of routing" (CPUs or CPs).
- The network configuration does not exceed project limits.
- The modules have loaded the configuration data containing the latest "knowledge" of the entire network configuration of the project.

  Reason: All modules participating in the network transition must receive the routing information defining the paths to other subnets.
- In your network configuration, the programming device/PC you want to use to establish a connection via network node must be assigned to the network it is physically connected to.
- The DP master module must be inserted at the IM151-7 CPU interface module and the CPU must be configured as DP master.
- If the MPI/DP interface is configured as DP slave, the routing function is only available when the DP interface is set active. In *STEP 7*, activate the "Test, Commissioning, Routing" checkbox in the properties of the DP interface. More information is available in the Programming with STEP 7 (http://support.automation.siemens.com/WW/view/en/18652056) manual or directly in the *STEP 7 Online Help*.

Reference

Additional information

- About configuring with *STEP 7* can be found in the Configuring Hardware and Connections in STEP 7 (http://support.automation.siemens.com/WW/view/en/45531110) manual
- On communication is available in the Communication with SIMATIC (http://support.automation.siemens.com/WW/view/en/25074283) System Manual.
- to SFCs can be found in the S7300-CPUs and ET 200-CPUs Instruction List (http://support.automation.siemens.com/WW/view/en/31977679).

  A detailed description is available in the *STEP 7 Online Help* or in the System and standard functions for S7-300/400 (http://support.automation.siemens.com/WW/view/en/1214574) Reference Manual.

3.3.8 Data set routing

Availability

The IM151-7 CPU interface module supports data set routing if a DP master module is inserted.

Routing and data set routing

Routing is the transfer of data beyond network boundaries. You can send information from a transmitter to a receiver across several networks.

Data record routing is an expansion of the "standard routing" and is used, for example, by *SIMATIC PDM*. The data sent through data set routing includes the parameters for the participating field devices (slaves) as well as device-specific information (e.g. setpoint values, limit values, etc.). The structure of the target address for data record routing depends on the data contents, i.e. the slave to which the data is to be sent.

With the PG, e.g. a parameter set that already exists on the field device can be read, edited and then sent back to the field device if the PG is not connected to the same PROFIBUS DP subnet as that of the field device.

The field devices themselves do not have to support data set routing, since they do not forward the information received.

Data set routing

The following figure shows the access of the PG to a variety of field devices. In doing so, the PG is connected via MPI to the IM 151-7 CPU interface module. The IM151-7 CPU interface module communicates via PROFIBUS with the field devices.

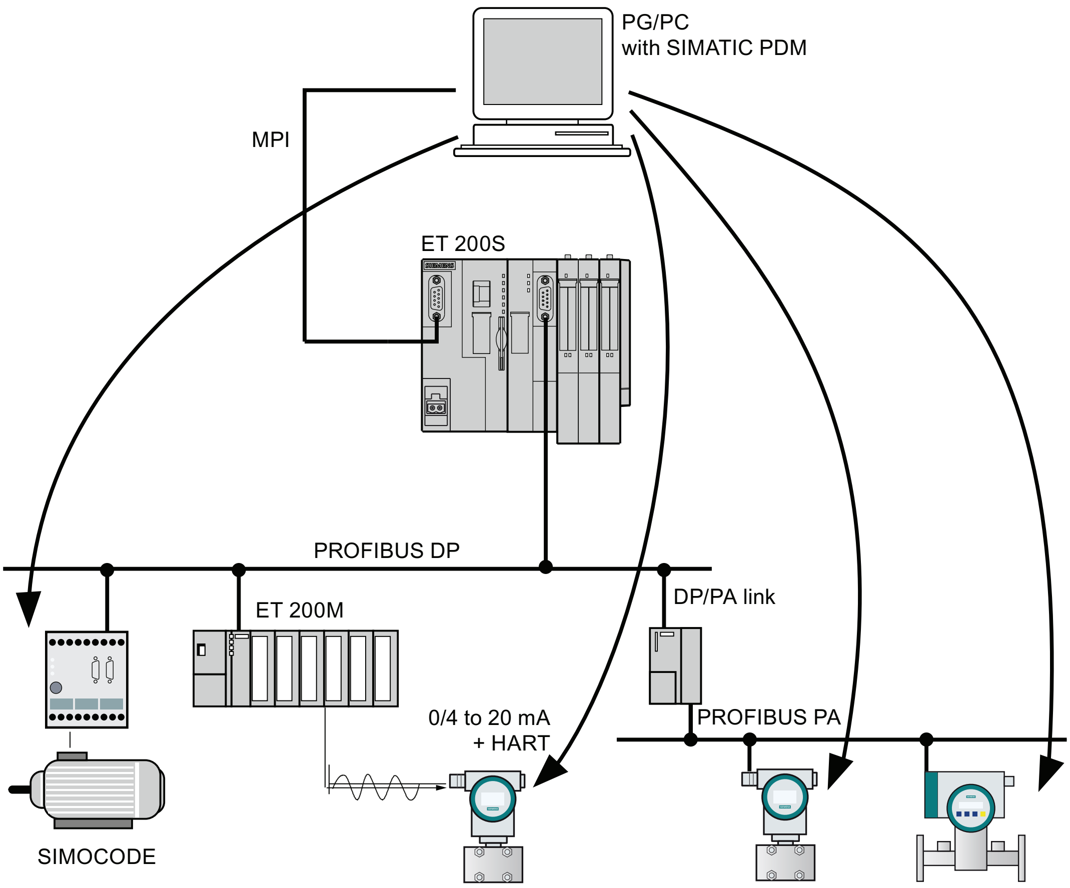


Figure 3-1 Data set routing with 151-7 CPU

See also

You can find additional information on *SIMATIC PDM* in the The Process Device Manager (http://support.automation.siemens.com/WW/view/en/21407212) Manual.

3.3.9 Clock synchronization

Introduction

The IM151-7 CPU interfaces support clock synchronization. The interface module can be parameterized for operation as time master (with default synchronization intervals) or as time slave.

Default: No clock synchronization

Setting the synchronization mode

The synchronization mode is set in HW Config in the properties dialog box of the IM151-7 CPU interface module or of the interface ("Clock" tab).

Interfaces

Clock synchronization is supported at the following interfaces:

- Combined MPI / DP interface, configured as MPI interface

  You can configure the interface module as

  – Time master or

  – Time slave or

  – "not synchronized".
- Combined MPI / DP interface, configured as DP interface (DP slave functionality)

  You can configure the interface module as

  – Time slave or

  – "not synchronized".
- With inserted DP master module (DP master functionality)

  You can configure the interface module as

  – Time master or

  – Time slave or

  – "not synchronized".

Note

The IM 151-7 CPU interface module may be the time slave on only one of these interfaces.

IM151-7 CPU as time slave

As a time slave, the IM151-7 CPU interface module receives synchronization frames from exactly one time master and accepts this time as its own internal time of the IM151-7 CPU.

IM151-7 CPU as time master

As time master, the IM151-7 CPU interface module broadcasts clock synchronization frames at parameterized synchronization intervals to synchronize other stations on the connected subnet.

Requirement: The clock of the IM151-7 CPU interface module is no longer in default state. The clock must have been set at least once.

Clock synchronization as **time master** starts:

- As soon as you initialize the time by means of SFC 0 "SET_CLK", or programming device function.
- Using another time master, if the IM151-7 CPU interface module is also parameterized via MPI/DP interface for operation as time slave.

Note

The clock of the IM151-7 CPU interface module is not set:

- in factory state
- after reset to factory settings by means of the mode selector switch
- after firmware updates

Example 1

If a DP master module is inserted and the IM151-7 CPU is configured as time slave at this interface, clock synchronization cannot be operated at the combined MPI / DP interface if it is configured as DP interface (because only time slave functionality could be set there, but time slave is only possible at one interface).

Example 2

If a DP master module is inserted and the IM151-7 CPU is configured as time slave at this interface, the IM151-7 CPU can be time master at the combined MPI / DP interface if the interface is configured as MPI interface.

3.3.10 Data consistency

Properties

A data area is consistent if it can be read or written to from the operating system as a consistent block. Data exchanged collectively between the stations should belong together and originate from a single processing cycle, that is, be consistent. If the user program contains a programmed communication function, for example, access to shared data with "X-SEND" / "XRCV", access to that data area can be coordinated by means of the "BUSY" parameter itself.

With PUT/GET functions

For S7 communication functions, such as PUT / GET or write / read via OP communication, which do not require a block in the user program on the IM151-7 CPU interface module (acting as a server), allowances must be made in the programming for the extent of the data consistency. The PUT / GET functions for S7 communication, or for read / write variables via OP communication, are executed at the IM151-7 CPU interface module´s cycle control point. In order to ensure a defined process interrupt reaction time, communication variables are copied consistently in blocks of up to 240 bytes to/from the user memory at the operating system's cycle control point. Data consistency is not guaranteed for larger data areas.

Note

If defined data consistency is required, the length of the communication variables in the user program of the IM151-7 CPU interface module must not exceed 240 byte.

3.4 S7 connections

3.4.1 S7 connection as communication path

An S7 connection is established when S7 modules communicate with one another. This S7 connection is the communication path.

Note

Global data communication as well as communication via PROFIBUS DP does not require S7 connections.

Each communication link requires S7 connection resources on the IM151-7 CPU interface module for the entire duration of this connection.

Thus, each IM151-7 CPU interface module is provided with a specific number of S7 connection resources. These are used by various communication services (PG / OP communication, S7 communication or S7 basic communication).

Connection points

An S7 connection between modules with communication capability is established between connection points. The S7 connection always has two connection points: The active and passive connection points:

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

Any module that is capable of communication can thus act as an S7 connection point. At the connection point, the established communication link always uses one S7 connection of the module concerned.

Transition point

If you use the routing functionality, the S7 connection between two modules capable of communication is established across a number of subnets. These subnets are interconnected via a network transition. The module that implements this network transition is known as a router. The router is thus the point through which an S7 connection passes.

Each IM 151-7 CPU interface module (with DP master module) can be a router for an S7 connection. You can establish a certain maximum number of routing connections. This does not limit the data volume of the S7 connections.

See also

Connection resources for routing (Page 50)

3.4.2 Assignment of S7 connections

There are several ways to allocate S7 connections on a communication-capable module:

- Reservation during configuration
- Assigning connections in the program
- Allocating connections during commissioning, testing and diagnostics routines
- Allocating connections for OCM services

Reservation during configuration

One connection resource each is automatically reserved on the IM151-7 CPU interface module for programming device and OP communication. Whenever you need more connection resources (for example, when connecting several OPs), configure this increase in the IM151-7 CPU interface module properties dialog box in *STEP 7*.

Assigning connections in the program

The user program establishes the connection with S7 basic communication. The connection is established by the operating system of the IM151-7 CPU interface module and the corresponding S7 connections are assigned.

Using connections for commissioning, testing and diagnostics

An active online function on the engineering station (programming device /PC with *STEP 7*) assigns S7 connections for programming device communication:

- An S7 connection resource for programming device communication which was reserved in your IM151-7 CPU interface module hardware configuration is assigned to the engineering station, that is, it just needs to be allocated.
- If all reserved S7 connection resources for programming device communication are allocated, the operating system automatically assigns a free S7 connection resource which has not yet been reserved. If no more connection resources are available, the engineering station cannot communicate online with the IM151-7 CPU interface module.

Allocating connections for OCM services

An online function on the HMI station (OP/TP/... with *WinCC*) is used for assigning S7 connection resources for the OP communication:

- An S7 connection resource for OP communication which was reserved in your IM151-7 CPU interface module hardware configuration is assigned to the HMI station, that is, it just needs to be allocated.
- If all reserved S7 connection resources for OP communication are allocated, the operating system automatically assigns a free S7 connection resource which has not yet been reserved. If no more connection resources are available, the HMI station cannot communicate online with the IM151-7 CPU interface module.

Chronological order in which S7 connection resources are assigned

When you program your project in *STEP 7*, the system generates parameter assignment blocks which are read by the modules in the startup phase. This allows the module's operating system to reserve or assign the relevant S7 connection resources. This means, for example, that OPs cannot access an S7 connection resource that has been reserved for programming device communication. If the IM151-7 CPU interface module has S7 connection resources that have not been reserved, these can be used freely. These S7 connection resources are allocated in the order they are requested.

Example

If there is only one free S7 connection left on the IM151-7 CPU interface module, you can still connect a programming device to the bus. The programming device can then communicate with the IM151-7 CPU interface module. However, the S7 connection will always be used if the PG is communicating with the IM151-7 CPU interface module. If you connect an OP to the bus while the programming device is not communicating, the OP can establish a connection to the IM151-7 CPU interface module. Since an OP maintains its communication link at all times, in contrast to the PG, you cannot subsequently establish another connection via the PG.

3.4.3 Distribution and availability of S7 connection resources

Distribution of connection resources

Table 3- 12 Distribution of connections

Communication service	Distribution
Programming device communication OP communication S7 basic communication	In order to avoid allocation of connection resources being dependent only on the chronological sequence in which various communication services are requested, connection resources can be reserved for these services. For PG and OP communication, at least one connection resource is reserved by default. The following table and the technical specifications for the IM151-7 CPU interface module contain the configurable S7 connections and the default setting. You "redistribute“ the connection resources by setting the relevant IM151-7 CPU interface module parameters in *STEP 7*.
S7 communication	Available connection resources that are not specially reserved for a service (programming device / OP communication, S7 basic communication) are used for this.
Routing PG functions	Together with the DP master module, the IM 151-7 CPU interface module has a number of connection resources available for routing purposes. These connections are available in addition to the connection resources. The subsection below shows the number of connection resources.
Global data communication	This communication service requires no S7 connection resources.
PROFIBUS DP	This communication service requires no S7 connection resources.

Availability of connection resources

Table 3- 13 Availability of connection resources

Interface module	Total number connection resources	Reserved for			Free S7 connections
		Programming device communication	OP communication	S7 basic communication	
IM151-7 CPU	12	1 to 11 default 1	1 to 11 default 1	0 to 10 default 0	Displays all non-reserved S7 connections as free connections.

3.4.4 Connection resources for routing

Number of connection resources for routing

In addition to the S7 connection resources, a maximum of four connection resources are available on the DP master module of the IM151-7 CPU interface module for the routing function. Routing is only possible if the DP master module is connected and configured.

Example for IM151-7 CPU

The IM 151-7 CPU interface module makes available 12 connection resources:

- Reserve two connection resources for programming device communication.
- Reserve two connection resources for OP communication.
- Reserve 3 connection resources for S7 basic communication.

This leaves 5 S7 connections available for any communication services, e.g. S7 communication, OP communication, etc.

In addition, 4 routing connections via the CPU are possible.

3.5 DPV1

New automation and process engineering tasks require the range of functions performed by the existing DP protocol to be extended. In addition to cyclical communication functions, acyclical access to non-S7 field devices is another important requirement of our customers, and was implemented in the EN 50170 standard. In the past, acyclic access was only possible with S7 slaves. The standard concerning distributed I/Os (EN 50170) has been further developed. All the changes concerning new DPV1 functions are included in IEC 61158/ EN 50170, volume 2, PROFIBUS.

Definition DPV1

The term DPV1 is defined as a functional extension of the acyclic services (to include new interrupts, for example) provided by the DP protocol.

Availability

Together with the DP master module you can operate the IM 151-7 CPU interface module as a DP master via the expanded DPV1 functionality.

Requirement for using the DPV1 functionality with DP slaves

For DPV1 slaves from other vendors, you will need a GSD file conforming to EN 50170, revision 3 or later.

Extended functions of DPV1

- Use of any DPV1 slaves from external vendors (in addition to the existing DPV0 and S7 slaves, of course).
- Selective handling of DPV1-specific interrupt events by new interrupt blocks.
- Reading/writing SFBs that conform to standards to the data record (although this can only be used for centralized I/O modules).
- User-friendly SFB for reading diagnostics.

Interrupt blocks with DPV1 functionality

Table 3- 14 Interrupt blocks with DPV1 functionality

OB	Functionality
OB 40	Process interrupt
OB 55	Status interrupt
OB 56	Update interrupt
OB 57	Vendor-specific interrupt
OB 82	Diagnostic interrupt

Note

You can now also use organization blocks OB40 and OB82 for DPV1 interrupts.

System blocks with DPV1 functionality

Table 3- 15 System function blocks with DPV1 functionality

SFB	Functionality
SFB 52	Read data record from DP slave or centralized I/O module
SFB 53	Write data record to DP slave or centralized I/O module
SFB 54	Read additional alarm information from a DP slave or a centralized I/O module in the relevant OB
SFB 75	Set any interrupts for intelligent slaves

Note

You can also use SFB 52 to SFB 54 for centralized I/O modules.

Reference

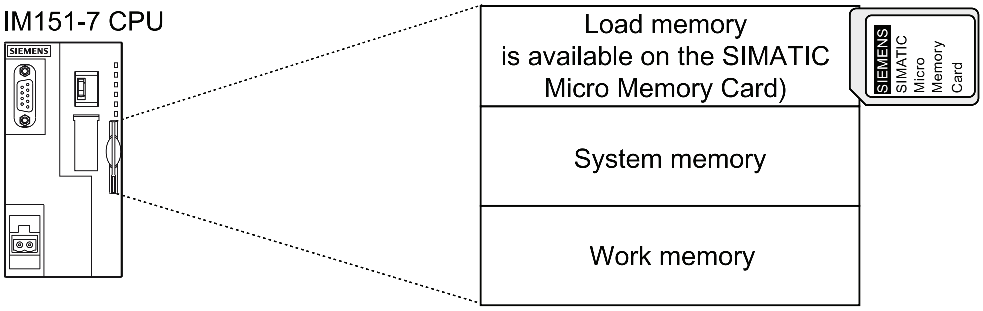
For additional information on the above blocks can be found in the System and Standard Functions for S7-300/400 (http://support.automation.siemens.com/WW/view/en/1214574) reference manual, or in the STEP 7 Online Help (http://support.automation.siemens.com/WW/view/en/1214574).

Memory concept 4

4.1 Memory areas and retentive memory

4.1.1 Memory areas of the IM151-7 CPU interface module

The three memory areas of your IM 151-7 CPU interface module



Load memory

The load memory is located on the SIMATIC Micro Memory Card. The size of the load memory corresponds exactly to the size of the SIMATIC Micro Memory Card. It is used to store code blocks, data blocks and system data (configuration, connections, module parameters, etc.). Blocks that are identified as non runtime-related are stored exclusively in load memory. You can also store all the configuration data for your project on the SIMATIC Micro Memory Card.

Note

User programs can only be downloaded and the IM151-7 CPU interface module can only be used if the SIMATIC Micro Memory Card is inserted in the interface module.

System memory

The system memory is integrated in the IM151-7 CPU interface module and cannot be extended.

It contains

- The address areas for address area memory bits, timers and counters
- The process image of the I/Os
- Local data

Work memory

The work memory is integrated in the IM151-7 CPU interface module and cannot be extended. It is used to run the code and process user program data. Programs only run in work memory and system memory.

4.1.2 Retentivity of load memory, system memory and RAM

Your IM151-7 CPU interface module is equipped with a maintenance-free retentive memory, i.e. its operation does not require a back-up battery. Data is kept in retentive memory across POWER OFF and restart (warm start).

Retentive data in load memory

Your program in the load memory is always retentive. It is stored on the SIMATIC Micro Memory Card, where it is protected against power failure or CPU memory restart.

The configuration data for the interface of the IM151-7 CPU interface module will be stored retentively in the load memory of an SDB.

Retentive data in system memory

In your configuration (Properties of IM151-7 CPU, Retentivity tab), specify which part of memory bits, timers and counters should be kept retentive and which of them are to be initialized with "0" on restart (warm restart).

The diagnostic buffer, MPI address (and transmission rate), and runtime meter data are generally stored in the retentive memory area on the IM151-7 CPU interface module. Retentivity of the MPI address and transmission rate ensures that your IM151-7 CPU interface module can continue to communicate, even after a power loss, memory reset, or loss of communication parameters (e.g. due to removal of the SIMATIC Micro Memory Card or deletion of communication parameters).

Only the last 100 entries in the diagnostics buffer are retentive with POWER OFF / POWER ON.

Retentive data in RAM

Therefore, the contents of retentive DBs are always retentive at restart and POWER OFF / POWER ON. Retentive data blocks can be uploaded to the work memory in accordance with the maximum limit allowed by the work memory.

The IM 151-7 CPU interface module also supports non-retentive DBs. Non-retentive DBs are initialized from the load memory with their initial values whenever a restart is performed or with POWER OFF / POWER ON. Non-retentive data blocks and code blocks can be loaded in accordance with the maximum work memory limit.

64 KB of RAM can be used for retentive data blocks in the IM 151-7 CPU interface module.

See also

Properties of the SIMATIC Micro Memory Card (Page 61)

4.1.3 Retentivity of memory objects

Retentive behavior of the memory objects

The table below shows the retentive behavior of memory objects during specific operating state transitions.

Table 4- 1 Retentive behavior of the memory objects

Memory object	Operating state transition		
	POWER OFF / POWER ON	STOP → RUN	Memory reset
User program / data (load memory)	X	X	X
• Retentive behavior of the DBs for the IM 151-7 CPU interface module	Can be set in the Properties of the DBs in *STEP 7* .		–
Bit memory, timers and counters configured as retentive data	X	X	–
Diagnostics buffers, operating hour counters	X[1]	X	X
MPI address, baud rate (or DP address, transmission rate of MPI / DP interface of the IM151-7 CPU, if it is parameterized as DP node)	X	X	X
X = retentive; – = not retentive [1] Only the last 100 entries in the diagnostics buffer are retained in the event of a POWER OFF / POWER ON.			

Retentive behavior of a DB with the IM 151-7 CPU interface module

For the IM 151-7 CPU interface module you can specify in *STEP 7* or via SFC 82 "CREA_DBL" (parameter ATTRIB -> NON_RETAIN bit), whether, in response to a POWER ON / OFF or RUN-STOP, a DB

- Keeps the actual values (retentive DB), or
- Accepts the initial values from load memory (non-retentive DB)

Table 4- 2 Retentive behavior of the DBs with the IM 151-7 CPU interface module

After a POWER OFF / POWER ON or restart of the IM 151-7 CPU interface module, the DB should...	
Receive the initial values (non-retentive DB)	**Retain the last actual values (retentive DB)**
Reason: After a POWER OFF / POWER ON and restart (STOP-RUN) of the IM 151-7 CPU interface module, the actual values of the DB are non-retentive. The DB receives the start values from load memory.	Reason: After a POWER OFF / POWER ON and restart (STOP-RUN) of the IM 151-7 CPU interface module, the actual values of the DB are retained.
Requirement in *STEP 7*: • The "Non-retain" check box must be activated in the block properties of the DB, or • a non-retentive DB was generated with SFC 82 "CREA_DBL" and the associated block attribute (ATTRIB -> NON_RETAIN bit).	Requirement in *STEP 7*: • The "Non-retain" check box must be activated in the block properties of the DB, or • A retentive DB was generated with SFC 82.

4.1.4 Address areas of system memory

The system memory of the IM 151-7 CPU interface module is broken down into address areas (refer to the table below). In a corresponding operation of your user program, you address data directly in the relevant address area.

Address areas of system memory

Table 4- 3 Address areas of system memory

Address areas	Description
Process image of inputs	At every start of an OB 1 cycle, the IM 151-7 CPU interface module reads the inputs from the input modules and saves the values to the process image of the inputs.
Process image of outputs	During its cycle, the program calculates the values for the outputs and writes these to the process image of outputs. At the end of the OB 1 cycle, the IM 151-7 CPU interface module writes the calculated output values to the output modules.
Bit memory	This area provides memory for saving the intermediate results of a program calculation.
Timers	Timers are available in this area.
Counters	Counters are available in this area.
Local data	Temporary data in a code block (OB, FB, FC) is saved to this memory area while the block is being edited.
Data blocks	See section Recipes (Page 68) and Measured value log files (Page 70).

Reference

The address areas of your IM 151-7 CPU interface module are listed in the S7-300-CPUs and ET 200-CPUs Instruction List (http://support.automation.siemens.com/WW/view/en/31977679).

I/O process image

When the user program addresses the input (I) and output (Q) address areas, it does not query the signal states of digital electronic modules; instead, it accesses a memory area in the IM 151-7 CPU interface module system memory. This particular memory area is the process image.

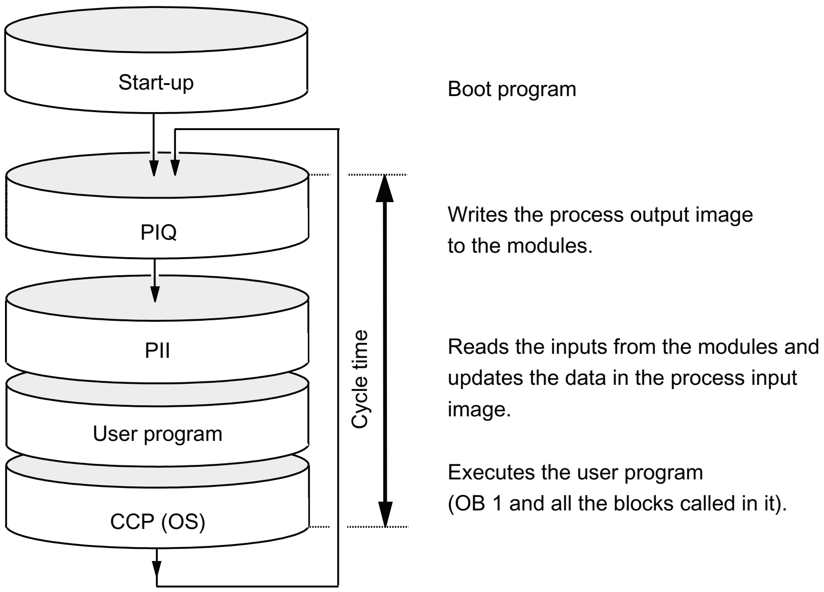
The process image is organized in two sections: The process image of inputs, and the process image of outputs.

Advantages of the process image

In contrast to direct access of the input / output modules, accessing the process image has the advantage that a consistent image of the process signals is made available to the IM 151-7 CPU interface module during cyclic program execution. When the signal status at an input module changes during program execution, the signal status in the process image is maintained until the image is updated in the next cycle. Moreover, because the process image is stored in the IM 151-7 CPU interface module system memory, access is significantly faster than direct access to the electronic modules.

Process image update

The operating system updates the process image periodically. The figure below shows the sequence of this operation within a cycle.



Variable process image

In *STEP 7*, you can set the size of the I/O process image to any value from 0 to 2048 bytes for the IM151-7 CPU interface module.

Please observe the following::

Note

Currently, the dynamic setting of the process image only affects its update at the cycle control point. That is, the process input image is only updated up to the set PII size with the corresponding values of the peripheral input modules existing within this address area, or the values of the process output image up to the set POI size are written to the peripheral output modules existing within this address area.

This set size of the process image is ignored with respect to the *STEP 7* commands used to access the process image (for example U I100.0, L IW200, = Q20.0, T AD150, or the corresponding indirect addressing commands). However, up to the maximum size of the process image (that is, up to I/O byte 2047), these commands do not return any synchronous access errors, but rather access the permanently available internal memory area of the process image.
The same applies to the use of actual parameters of block calls from the I/O area (area of the process image).

Particularly if these process image limits were changed, you should check to which extent your user program continues to access the process image in the area between the set and the maximum process image size. If such access is to continue to take place, this means that inputs on the I/O module that change may not be detected or that outputs may not really be written to the output module without an error message being generated.

Local data

Local data store:

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

Temporary Variables

When you create blocks, you can declare temporary variables (TEMP) which are only available during block execution and then overwritten again. These local data have fixed length in each OB. Local data must be initialized prior to the first read access. Each OB also requires 20 bytes of local data for its start information.

The IM 151-7 CPU interface module has memory for storing temporary variables (local data) of recently executed blocks. This memory is divided among the priority classes into partitions of equal size. Each priority class has its own local data area.

> **CAUTION**
>
> All temporary variables (TEMP) of an OB and its nested blocks are stored in local data. When using complex nesting levels for block processing, you may cause an overflow in the local data area.
> The IM 151-7 CPU interface module will change to STOP mode if the permitted length of the local data for a priority class is exceeded.
> Make allowances for local data space required for synchronous error OBs. This is assigned to the respective triggering priority class.

See also

Retentivity of load memory, system memory and RAM (Page 54)

4.1.5 Properties of the SIMATIC Micro Memory Card

The SIMATIC Micro Memory Card as a memory module for the IM 151-7 CPU interface module

Your IM151-7 CPU interface module uses a SIMATIC Micro Memory Card as memory module. It can be used as load memory or as a portable storage medium.

Note

The IM151-7 CPU interface module requires the SIMATIC Micro Memory Card for operation.

The following data are stored on the SIMATIC Micro Memory Card:

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

Note

You can either store user and configuration data or the operating system on the SIMATIC Micro Memory Card.

Properties of a SIMATIC Micro Memory Card

The SIMATIC Micro Memory Card ensures maintenance-free and retentive operation of the IM 151-7 CPU interface module .

SIMATIC Micro Memory Card copy protection

Your SIMATIC Micro Memory Card has an internal serial number that implements an MMC copy protection. You can read this serial number from the SSL partial list 011C$_H$ index 8 using SFC 51 "RDSYSST". If the reference and actual serial number of your SIMATIC Micro Memory Card are not the same, program a STOP command in a know-how-protected module, for example.

Reference

Additional information

- on the *SSL partial list* refer to the S7-300-CPUs and ET 200-CPUs Instruction list (http://support.automation.siemens.com/WW/view/en/31977679) or the System and standard functions for S7-300/400 (http://support.automation.siemens.com/WW/view/en/1214574) Reference Manual.
- For memory reset of the IM151-7 CPU interface module go to section Memory reset of the IM151-7 CPU interface module using the mode selector switch (Page 93).

Useful life of a SIMATIC Micro Memory Card

The life of an SIMATIC Micro Memory Card depends mainly on the following factors:

1. The number of delete or programming cycles
2. External influences, such as ambient temperature.

At ambient temperatures up to 60 °C, up to 100,000 delete/write operations can be performed on a SIMATIC Micro Memory Card.

⚠ CAUTION
To prevent data losses, do not exceed this maximum of delete/write operations.

See also

Retentivity of load memory, system memory and RAM (Page 54)
Operating and display elements of the IM151-7 CPU interface module (Page 17)

4.2 Memory functions

4.2.1 General: Memory functions

Memory functions

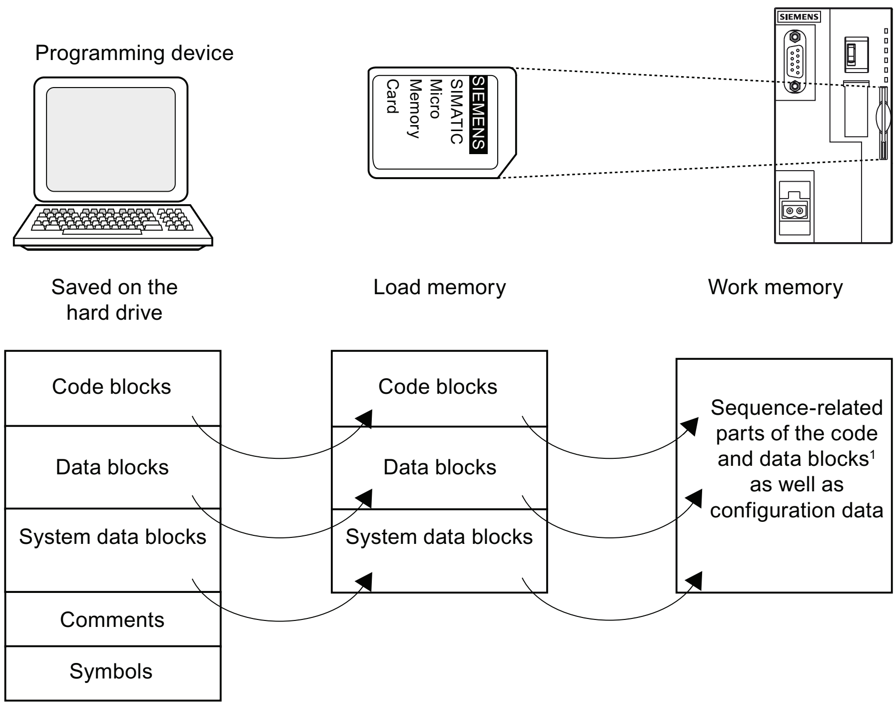
Memory functions are used to generate, modify or delete entire user programs or specific blocks. You can also ensure that your project data are retained by archiving these. If you created a new user program, use a programming device/PC to download the complete program to the SIMATIC Micro Memory Card.

4.2.2 Downloading user programs via SIMATIC Micro Memory Card to the IM 151-7 CPU interface module

User program download

The entire user program is downloaded from your PG / PC to the IM 151-7 CPU interface module via the SIMATIC Micro Memory Card. The previous content of the SIMATIC Micro Memory Card is deleted in the process. Blocks use the load memory area as specified under "Load memory requirements" in "General block properties".

The figure shows the load and work memory of the IM 151-7 CPU interface module.



[1]: If not all of the work memory area is retentive, the retentive part is indicated in the *STEP 7* module status as retentive memory. You cannot run the program until all the blocks are downloaded.

Note

This function is only permitted if the IM 151-7 CPU interface module is in STOP mode. Load memory is cleared if the load operation could not be completed due to power loss or illegal block data.

4.2.3 Handling blocks

4.2.3.1 Encryption of blocks

Important information

Note

Supported blocks

S7 Block Privacy can be used to encrypt only function blocks (FBs) and functions (FCs).

Once blocks have been encrypted in *STEP 7* they can no longer be edited or monitored. Nor can any test or commissioning functions, such as status blocks or breakpoints, be performed.

Requirement

The "S7 Block Privacy" extension package supplied with *STEP 7* must be installed. Only by this means can blocks be hard encrypted.

General procedure

In order to encrypt the blocks, proceed as follows:

1. In SIMATIC Manager, right-click the block container and select "Block Privacy ...".
2. The "S7-Block Privacy" tool is launched.
3. In the block browser of "S7-Block Privacy", select the desired block (multiple selections can be made).
4. Right click on the block to be encrypted, and select "Encrypt block...". The "Block encryption" dialog box opens.
5. Select whether decompilation information should also be encrypted.

   Note

   If you deactivate the check box, there is no way the block can be decompiled.

   ---

6. Enter a code of at least 12 characters in both fields. Make sure the code is securely stored. Press the "OK" button to start the encryption.

   Result: Your block is now encrypted. This is indicated by the following symbols:

 Block that can be decompiled

 Block that cannot be decompiled

---

Note

Run time for the command

Typically the run time for the command is extended because the encrypted blocks could not be executed in a fully optimized manner. The final cycle time can only be determined with encrypted blocks.

Note

Extended run times for POWER ON/memory reset/download

The ramp-up time for the CPU, the time required for memory reset and the loading time of blocks can be extended significantly.

Additional information

For more information please refer to the *STEP 7 Online Help* under "S7 Block Privacy".

4.2.3.2 Download of new blocks or delta downloads

There are two ways to download additional user blocks or download deltas:

- Download of blocks: You have already created a user program and downloaded it to the SIMATIC Micro Memory Card in the IM 151-7 CPU interface module. You then want to add new blocks to the user program. In this case you do not need to reload the entire user program to the MCC. Instead you only need to download the new blocks to the SIMATIC Micro Memory Card (this reduces the download times for highly complex programs).
- Delta download: In this case, you only download the deltas in the blocks of your user program. In the next step, perform a delta download of the user program, or only of the changed blocks to the SIMATIC Micro Memory Card, using the programming device/PC.

> **WARNING**
>
> The delta down of block / user programs overwrites all data stored under the same name on the SIMATIC Micro Memory Card.

The data of dynamic blocks are transferred to RAM and activated after the block is downloaded.

4.2.3.3 Uploading blocks

In contrast to downloading, uploading involves the transfer of individual blocks or a complete user program from the IM 151-7 CPU interface module to the programming device / PC. In doing so, the blocks have the content of the last download in the IM 151-7 CPU interface module. Dynamic DBs form the exception, because their actual values are transferred. Uploading blocks or the user program from the IM 151-7 CPU interface module with *STEP 7* does not affect the memory assignment of the IM 151-7 CPU interface module.

4.2.3.4 Deleting blocks

When you delete a block, it is deleted from load memory. In *STEP 7*, you can also delete blocks with the user program (DBs also with SFC 23 "DEL_DB"). RAM used by this block is released.

4.2.3.5 Compressing blocks

When data are compressed, gaps which have developed between memory objects in load memory/RAM as a result of load/delete operations will be eliminated. This releases free memory in a continuous block. You can compress both in STOP mode as well as in RUN mode of the IM 151-7 CPU interface module.

4.2.3.6 Promming (RAM to ROM)

When writing the RAM content to ROM, the actual values of the DBs are transferred from RAM to load memory to form the start values for the DBs.

Note

This function is only permitted if the IM 151-7 CPU interface module is in STOP mode. Load memory is cleared if the function could not be completed due to power loss.

4.2.4 CPU memory reset and restart

Memory reset

After inserting or removing a SIMATIC Micro Memory Card, a complete memory reset restores the IM 151-7 CPU interface module to defined conditions in order to make a restart (warm start) possible. When resetting the IM 151-7 CPU interface module, the memory management of the IM 151-7 CPU interface module is reestablished. Blocks in load memory are retained. All dynamic runtime blocks are transferred once again from load memory to RAM, in particular to initialize the data blocks in RAM (restore initial values).

Restart (warm start)

- All retentive DBs retain their current values. Non-retentive DBs are reset to their initial values.
- The values of all retentive M, C, T are retained.
- All non-retentive user data are initialized:
  - M, C, T, I, O with "0"
- All run levels are initialized.
- The process images are deleted.

Reference

Also read section Memory reset of the IM151-7 CPU interface module using the mode selector switch (Page 93).

4.2.5 Recipes

Introduction

A recipe represents a collection of user data. You can implement a simple recipe concept using static DBs. In this case, the recipes should have the same structure (length). One DB should exist per recipe.

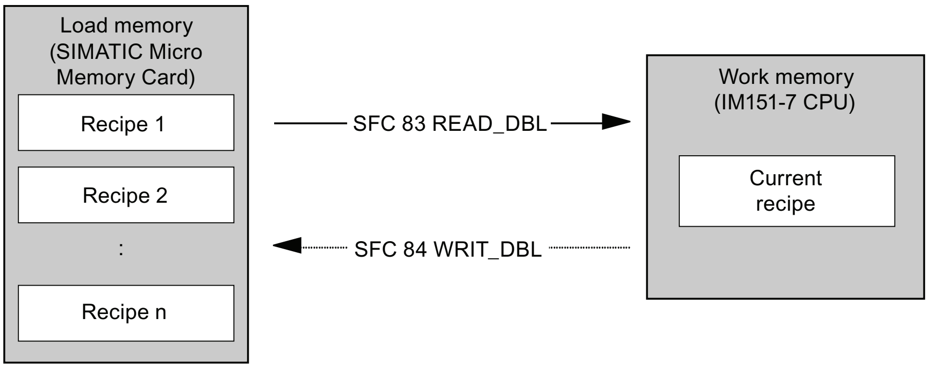
Processing sequence

Recipe is written to load memory:

- The individual data records of the recipes are created as static DBs in *STEP 7* and then downloaded to the IM 151-7 CPU interface module. Therefore, recipes only use load memory, rather than RAM.

Working with recipe data:

- SFC83 "READ_DBL" is called in the user program to copy the data record of a current recipe from the DB in load memory to a static DB that is located in work memory. As a result, the RAM only has to accommodate the data of one record. The user program can now access data of the current recipe. The figure below shows how to handle recipe data:



Saving a modified recipe:

- The data of new or modified recipe data records generated during program execution can be written to load memory. To do this, call SFC 84 "WRIT_DBL" in the user program. The data written to load memory are portable and retentive on Memory reset. You can backup modified records (recipes) by uploading and saving these in a single block to the programming device/PC.

Note

Active system functions SFC82 to 84 (active access to the SIMATIC Micro Memory Card) have a distinct influence on programming device functions (for example, block status, variable status, download block, upload, open). This typically reduces performance (compared to passive system functions) by a factor of 10.

Note

To prevent data losses, do not exceed this maximum of delete/write operations.

CAUTION

Data on a SIMATIC Micro Memory Card can be corrupted if you remove the card while it is being accessed by a write operation. In this case, you may have to delete the SIMATIC Micro Memory Card on your PG, or format the card in the IM151-7 CPU interface module.

Never remove a SIMATIC Micro Memory Card in RUN mode. Always remove it when power is off or when the IM151-7 CPU interface module is in STOP state, and when the PG is not writing to the card. If the CPU is in STOP mode and you cannot not determine whether or not a PG is writing to the card (e.g. load/delete block), disconnect the communication lines.

4.2.6 Measured value log files

Introduction

Measured values are generated when the IM151-7 CPU interface module executes the user program. These measured values are to be logged and analyzed.

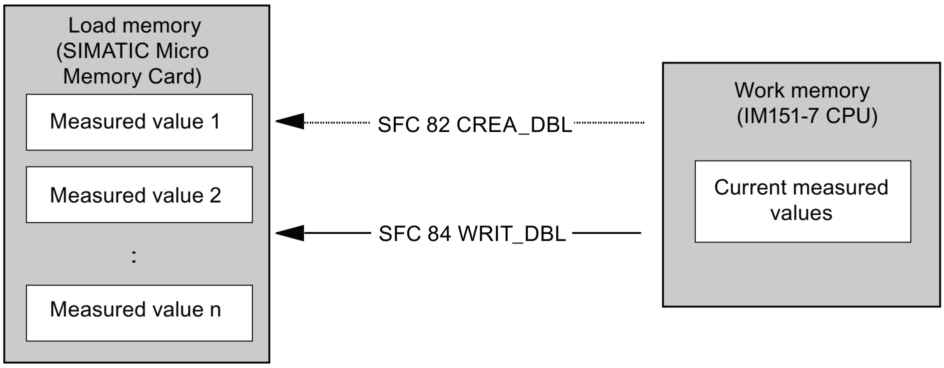
Processing sequence

Acquisition of measured values:

- The IM 151-7 CPU interface module writes all measured values to a DB (for alternating backup mode in several DBs) which is located in the work memory.

Measured value logging:

- Before the data volume can exceed work memory capacity, you should call SFC 84 "WRIT_DBL" in the user program to swap measured values from the DB to load memory. The figure below shows how to handle measured value log files:



- You can call SFC 82 "CREA_DBL" in the user program to generate new (additional) static DBs in load memory which do not require RAM space.

Reference

For additional information on the block SFC 82 can be found in the System and Standard Functions for S7-300/400 (http://support.automation.siemens.com/WW/view/en/1214574) reference manual, or in the STEP 7 Online Help (http://support.automation.siemens.com/WW/view/en/1214574).

Note

SFC 82 is terminated and an error message is generated if a DB already exists under the same number in load memory and/or work memory.

The data written to load memory are portable and retentive on Memory reset.

Evaluation of measured values:

- Measured value DBs saved to load memory can be uploaded and evaluated by other communication partners (programming device, PC, for example).

Note

Active system functions SFC82 to 84 (active access to the SIMATIC Micro Memory Card) have a distinct influence on programming device functions (for example, block status, variable status, download block, upload, open). This typically reduces performance (compared to passive system functions) by a factor of 10.

Note

With the IM 151-7 CPU interface module you can also generate non-retentive DBs using SFC 82 (parameter ATTRIB -> NON_RETAIN bit).

Note

To prevent data losses, do not exceed this maximum of delete/write operations.

4.2.7 Backup of project data to SIMATIC Micro Memory Card

Function principle

Using the **Save project to Memory Card** and **Fetch project from Memory Card** functions, you can save all project data to a SIMATIC Micro Memory Card, and retrieve these at a later time. For this operation, the SIMATIC Micro Memory Card can be located in the IM 151-7 CPU interface module or in the programming adapter of a programming device or PC.

Project data is compressed before it is saved to a SIMATIC Micro Memory Card, and uncompressed on retrieval.

Note

In addition to project data, you may also have to store your user data on the MMC. You should therefore first select a SIMATIC Micro Memory Card with sufficient free memory.

A message warns you if the memory capacity on your SIMATIC Micro Memory Card is insufficient.

The volume of project data to be saved corresponds with the size of the project's archive file.

Note

For technical reasons, you can only transfer the entire contents (user program and project data) using the **Save project to memory card** action.

Mounting and connecting 5

5.1 Content

Where can I find what information?

You will find comprehensive information about fitting and connecting an ET 200S in the relevant sections of the ET 200S Distributed I/O Device (http://support.automation.siemens.com/WW/view/en/1144348) Operating Instructions.

The following sections will show you the differences and special features associated with using an ET 200S with the IM 151-7 CPU interface module.

5.2 Installing the IM151-7 CPU interface module

Requirement

The mounting rail is installed.

Procedure

1. Install the IM 151-7 CPU interface module.
2. Mount the required terminal modules.

   Note

   Note the installation sequence

   If you wish to extend the IM 151-7 CPU interface module with an optional DP master module, you must first of all install the DP master module before you install the required terminal modules.

   ---

3. Mount the terminating module.

Reference

The mounting is described in the ET 200S Distributed I/O System (http://support.automation.siemens.com/WW/view/en/1144348) Operating Instructions.

5.3 Connecting the IM151-7 CPU interface module

Introduction

You connect the supply voltage and PROFIBUS DP to the IM151-7 CPU interface module.

Requirements

- The IM151-7 CPU interface module is installed on the mounting rail.
- Wire the interface module with the supply voltage switched off.
- Observe the wiring rules in the ET 200S Distributed I/O System (http://support.automation.siemens.com/WW/view/en/1144348) Operating Instructions.

Required tools

3 mm screwdriver

Power supply

You may only use SELV / PELV-type power supply units with a guaranteed electrically isolated extra-low voltage (≤ 60 V DC).

Required accessories

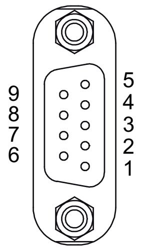
- Cable with maximum 2.5 mm^2 conductor cross section for the supply voltage
- PROFIBUS DP bus connector
  Suitable bus connectors are listed in section Order numbers of accessories (Page 194).

Connecting PROFIBUS DP

Connect the PROFIBUS DP as follows:

1. Use a pre-fabricated PROFIBUS cable.
2. Plug the bus connector into the PROFIBUS DP socket.
3. Screw the bus connector into the connection socket.

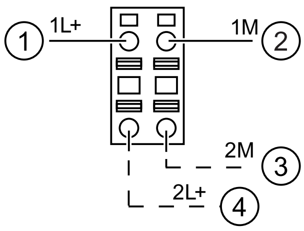
Pin assignment for the bus connector:

View of the connection socket	Terminal	Signal	Designation
 RS 485 interface	1	–	–
	2	M24	external 24 V supply
	3	RxD/TxD-P	Data line B
	4	RTS	Request To Send
	5	M5V2	Data reference potential (station)
	6	P5V2	Supply plus (station)
	7	P24	external 24 V supply
	8	RxD / TxD-N	Data line A
	9	–	–

Connecting the supply voltage

The voltage supply is over a connecting plug. When it ships, this is connected to the connection for the supply voltage on the IM151-7 CPU interface module.
The connecting plug makes it possible to loop the voltage supply uninterrupted.

Pin assignment of the interface:

	①	24 VDC (red)
	②	Ground (blue)
	③	Ground (blue)
	④	24 VDC (red)

Connect the supply voltage as follows:

1. Strip the wires for the supply voltage to 10 mm.
2. Insert the individual cables into the spring-loaded terminal (round openings) of the cable connector.
3. Insert the wired connector onto the 24 VDC terminal of the IM151-7 CPU interface module.
4. Please ensure that there is sufficient strain relief.

5.4 Installing and connecting the DP master module

If you wish to extend the IM151-7 CPU interface module with an optional DP master module, you can use the IM151-7 CPU interface module as a DP master. Connect the PROFIBUS DP to the DP master module.

Requirements

- The mounting rail has been fitted (see the ET 200S Distributed I/O Device (http://support.automation.siemens.com/WW/view/en/1144348) Operating Instructions).
- The IM151-7 CPU interface module is installed on the mounting rail (see section Installing the IM151-7 CPU interface module (Page 73)).

> **Note**
>
> You should install the required terminal module only having first installed the DP master module.

Installing the DP master module

1. Suspend the DP master module in the mounting rail to the right of the IM151-7 CPU interface module.
2. Rotate the DP master module to the back until it engages.
3. Slide the DP master module to the left until it audibly engages with the IM151-7 CPU interface module.

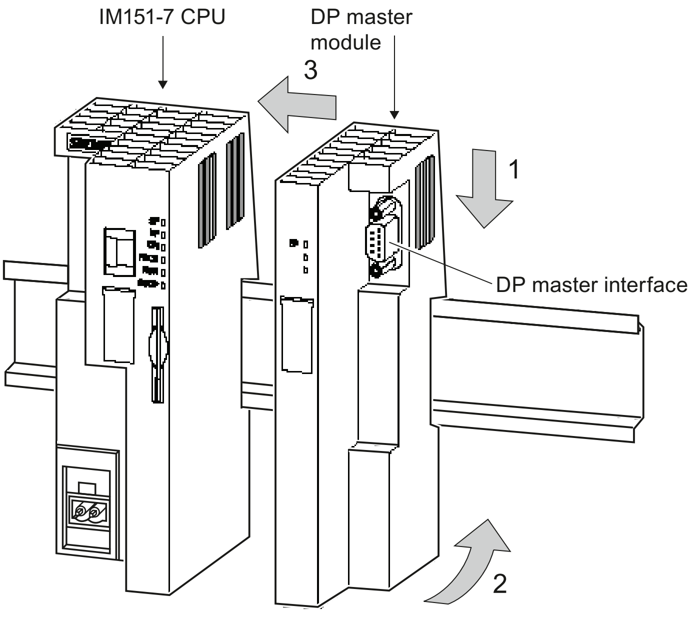


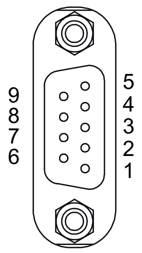
Figure 5-1 Installing the DP master module

Connecting PROFIBUS DP

Connect the PROFIBUS DP as follows:

- Use a pre-fabricated PROFIBUS cable.
- Insert the bus connector in the X1 connection socket on the DP master module.
- Screw the bus connector into the connection socket.

Pin assignment for the bus connector:

View of the connection socket	Terminal	Signal	Designation
	1	–	–
	2	–	–
	3	RxD / TxD-P	Data line B
	4	RTS	Request To Send
	5	M5V2	Data reference potential (station)
	6	P5V2	Supply plus (station)
	7	–	–
RS 485 interface	8	RxD / TxD-N	Data line A
	9	–	–

Addressing 6

6.1 Addressing the I/O modules

6.1.1 Slot-oriented addressing of the centralized I/O modules

Slot-oriented addressing

A slot-orientated addressing is available only for the centralized I/O of the IM151-7 CPU interface module. If the IM151-7 CPU interface module is started up without a configuration loaded, the I/O modules are addressed by slot by default.

This is the digital or analog address set by default according to the type of I/O module (see table below).

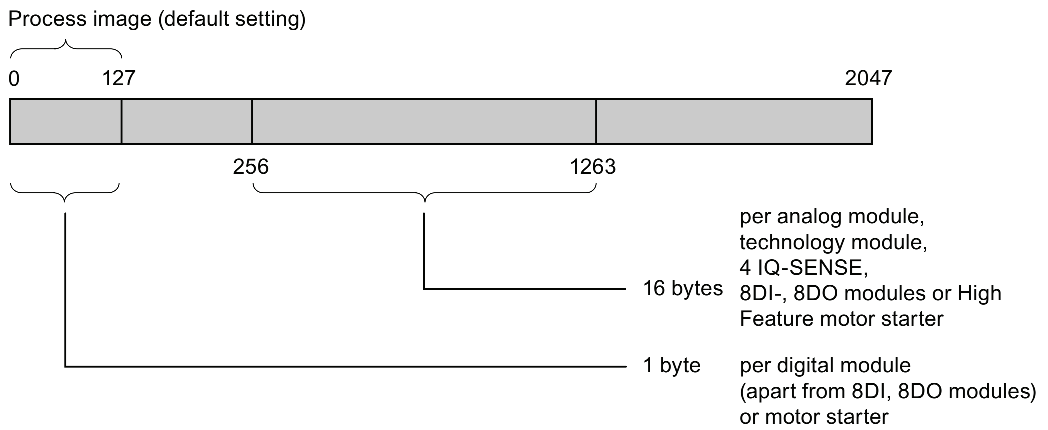


Figure 6-1 Structure of the default address area

Slot assignment

The figure below shows an ET 200S configuration with digital and analog electronic modules as well as the technology modules and slot assignment.

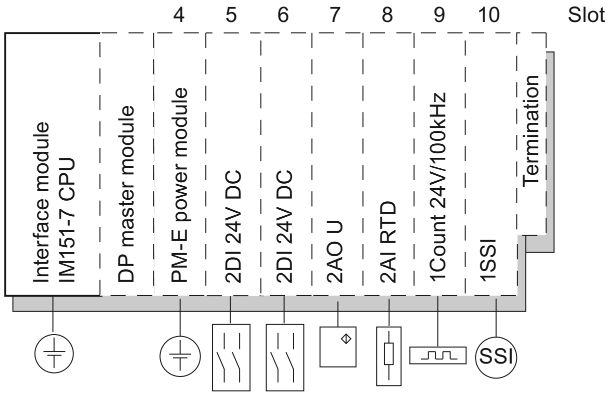


Figure 6-2 Slots for the ET 200S

Address assignment

Depending on the slot, 1 byte is reserved in the address ranges of the IM151-7 CPU interface module for digital I/Os and motor starters, and 16 bytes is reserved for analog I/Os, technology modules, 4 IQ-SENSE and High Feature motor starters (up to 63 I/O modules).

The table below indicates the default address assignment for analog and digital modules per slot for slot-oriented addressing.

Table 6- 1 Default address assignment for centralized I/O modules in an ET 200S with IM151-7 CPU

Reserved address range	Slot number									
	1	2	3	4	5	6	7	8	…	66
Digital modules (except 8DI, 8DO modules), motor starters	IM151-7 CPU [1]			-	1	2	3	4	…	62
Analog modules, technology modules, 4 IQ SENSE, 8DI, 8DO modules, High Feature motor starters				-	272 to 287	288 to 303	304 to 319	320 to 335	…	1248 to 1263
Power modules [2]				256	272	288	304	320	…	1248

[1] with X1 as MPI/DP interface and X2 as DP master interface

[2] Diagnostics addresses (no user data)

Note

The following digital modules are treated as analog or TF modules for default address assignment purposes:

- 6ES7131-4BF00-0AA0
- 6ES7131-4BF50-0AA0
- 6ES7132-4BF00-0AA0
- 6ES7132-4BF00-0AB0
- 6ES7132-4BF50-0AA0
- 6ES7131-4RD00-0AB0
- 6ES7131-4RD02-0AB0

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.

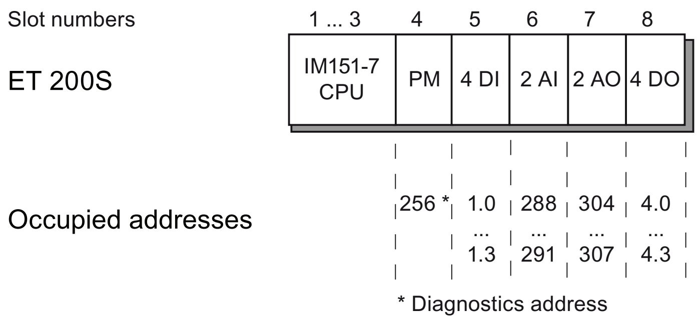


Figure 6-3 Example of address assignment for I/O modules

6.1.2 User-oriented addressing of the I/O Modules

User-oriented addressing

User-oriented module addressing is possible with both centralized and distributed I/Os.

User-oriented addressing means you can freely select

- Input addresses for modules and
- Output addresses for modules

within the range 0 to 2047 with byte-level granularity independently of one another. Assign the addresses in *STEP 7*. Specify the module start address that forms the basis for all other addresses of the module.

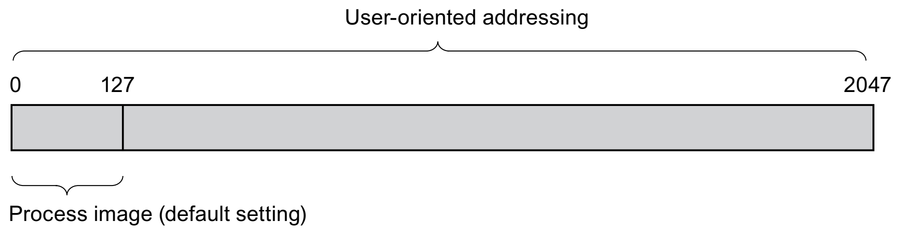


Figure 6-4 Structure of the address area for user-oriented addressing

---

Note

If you are using PROFIBUS DP field devices, you must always configure the hardware in *STEP 7* HW Config. User-oriented addressing of modules is used automatically, and there is no fixed slot addressing.

Advantage

Advantages of user-oriented addressing:

- You can make the best possible use of the available address spaces because there are no "address gaps" between the modules.
- When creating standard software, you can specify addresses that are independent of the configuration of the ET 200S station.

6.2 Addressing on PROFIBUS DP

Overview

The relevant DP slaves must first be brought into service on the PROFIBUS DP before the distributed I/Os can be addressed from the user program.

During this commissioning process:

- PROFIBUS addresses are assigned to the slaves
- Address ranges are assigned to the input / output modules or slots so that they can be addressed from the user program. Slots without user data are given a diagnostics address.

Additional information on commissioning the IM151-7 CPU interface module

- as DP slave (I slave) is available in the section Commissioning IM151-7 CPU as DP slave (Page 112)
- as DP master is available in the section Commissioning IM151-7 CPU with DP master module as DP master (Page 119).

User-oriented addressing of the distributed PROFIBUS I/Os

You must use user-oriented addressing for the distributed PROFIBUS DP I/Os.

Information can be found in the User-oriented addressing of the I/O modules (Page 82) section.

Addressing consistent user data areas

The table below illustrates the points to consider with respect to communication in a PROFIBUS DP master system if you want to transfer I/O areas with "Total length" consistency.

For 1 to 32 byte data consistency on the PROFIBUS DP:
The address range of consistent data in the process image is automatically updated.
To read and write consistent data, you can also use SFC 14 "DPRD_DAT" and SFC 15 "DPWR_DAT". If the address range of consistent data is not in the process image, you must use SFC 14 and SFC 15 to read and write consistent data.
The length in the SFC must tally with the length of the programmed area when accessing areas with "Total length" consistency.
Direct access to consistent areas is also possible (e.g. L PEW or T PAW).
In a PROFIBUS DP system you can transfer up to 32 bytes of consistent data.

Commissioning 7

7.1 Overview

This section contains important notes on commissioning which you should strictly observe in order to avoid injury or damage to machines.

Note

Your commissioning phase is determined primarily by your application, so we can only offer you general information, without claiming completeness of this topic.

Reference

Note the information about commissioning provided in the descriptions of your system components and devices.

7.2 Commissioning procedure

7.2.1 Procedure: Commissioning the hardware

Hardware requirements

- ET 200S is installed
- ET 200S is connected

In case of networking the ET 200S with MPI or PROFIBUS:

- The MPI/PROFIBUS addresses are configured
- the terminating resistors on the segments are enabled

Recommended procedure: Hardware

Due to its modular structure and the many different expansion options, an ET 200S can be very large and complex. It is therefore not a good idea to switch it on for the first time with all the modules installed. Rather, we recommend a step-by-step commissioning procedure.

We recommend the following initial commissioning procedure for an ET 200S:

Table 7- 1 Recommended commissioning procedure: Hardware

Activity	Remarks	Information on this can be found in section
An installation and wiring check according to checklist	-	Commissioning check list (Page 88)
Disconnecting drive aggregates and control elements	This prevents negative effects on your system as a result of program errors. Tip: By redirecting data from your outputs to a data block, you can always check the status at the outputs	-
Preparing IM151-7 CPU	Connecting a programming device	Connecting the programming device (PG) (Page 100)
Start up the power supply and ET 200S with the IM151-7 CPU and check the LEDs	Commission the power supply and the ET 200S with inserted IM151-7 CPU interface module.	Initial power on (Page 92)
	Check the LED displays of the IM151-7 CPU interface module.	Debugging functions, diagnostics and troubleshooting (Page 141)
Reset IM151-7 CPU memory and check the LEDs	-	Memory reset of the IM151-7 CPU interface module using the mode selector switch (Page 93)
Commission the rest of the modules	Plug the modules in according to the configuration and commission them.	ET 200S Distributed I/O System (http://support.automation.siemens.com/WW/view/en/1144348) Operating Instructions

DANGER

Proceed step-by-step. Do not go to the next step unless you have completed the previous one without error / error message.

Reference

Important information is available in section Debugging functions, diagnostics and troubleshooting (Page 141).

See also

Procedure: Software commissioning (Page 87)

7.2.2 Procedure: Software commissioning

Requirements

- You have installed and connected your ET 200S with IM151-7 CPU.
- In order to utilize the full functionality of your IM151-7 CPU interface module, you require *STEP 7* as of V5.2 + SP1 + HSP219 or V5.5 + SP1.V5.5.
- In case of networking the ET 200S with MPI or PROFIBUS:
  - The MPI/PROFIBUS addresses are configured
  - The terminating resistors on the segment limits are enabled.

> **Note**
>
> Please observe the procedure for commissioning the hardware.

Recommended procedure: Software

Table 7- 2 Recommended commissioning procedure: Software

Activity	Remarks	Information can be found ...
• Switch on the programming device and run SIMATIC Manager • Transferring the configuration and program to the IM151-7 CPU interface module	-	in the Programming with STEP 7 (http://support.automation.siemens.com/WW/view/en/18652056) Manual
Debugging the I/Os	Helpful functions are here: • Monitoring and modifying variables • Testing with program status • Forcing • Controlling the outputs in STOP mode (PO enable) **Tip:** Test the signals at the inputs and outputs.	• in the *Programming with STEP 7* Manual • in section Debugging functions, diagnostics and troubleshooting (Page 141)
Commissioning PROFIBUS DP	-	in section Commissioning PROFIBUS DP (Page 110)
Connecting the outputs	Commissioning the outputs successively.	-

> **DANGER**
>
> Proceed step-by-step. Do not go to the next step unless you have completed the previous one without error / error message.

Reaction to errors

React to errors as follows:

- Check the system with the help of the check list in the chapter below.
- Check the LED displays on all modules. The meaning is described in the ET 200S Distributed I/O Device (http://support.automation.siemens.com/WW/view/en/1144348) Operating Instructions.
- If required, remove individual components to trace the error.

Reference

Important information is available in section Debugging functions, diagnostics and troubleshooting (Page 141).

See also

Procedure: Commissioning the hardware (Page 85)

7.3 Commissioning check list

Introduction

Once you have installed and wired up your ET 200S, we advise you to check all the previous steps once again.

The following tables give you instructions in the form of a checklist for checking your ET 200S. They also provide cross-references to sections containing additional information on the relevant topic.

Mounting rail

The points to be checked are listed in the ET 200S Operating Instructions (http://support.automation.siemens.com/WW/view/en/1144348)	in chapter
Is the rail mounted firmly to the wall, in the frame or in the cabinet?	*Installing*
Have you maintained the free space required?	*Installing*

Concept of grounding and chassis ground

The points to be checked are listed in the ET 200S Operating Instructions (http://support.automation.siemens.com/WW/view/en/1144348)	in chapter
Have you established a low-impedance connection (large surface, large contact area) to ground potential?	*Wiring and assembly*
Is the profile rail properly connected to reference potential and ground potential (direct electrical connection or ungrounded operation)?	*Wiring and assembly*
Are all grounding points of electrically connected measuring instruments and of the load power supply units connected to reference potentials?	*Appendix*

Module installation and wiring

The points to be checked are listed in the ET 200S Operating Instructions (http://support.automation.siemens.com/WW/view/en/1144348)	in chapter
Are all the terminal modules, including the terminating module, installed correctly?	*Installing*
Are all the terminal modules wired up correctly?	*Wiring and assembly*
Are all the power modules, electronic modules, ... correctly connected?	*Wiring and assembly*

7.4 Commissioning the modules

7.4.1 Inserting/Replacing a SIMATIC Micro Memory Card

SIMATIC Micro Memory Card (MMC) as memory module

Your IM151-7 CPU interface module uses a SIMATIC Micro Memory Card as memory module. You can use the SIMATIC Micro Memory Card as a load memory or a portable data medium.

Note

You must have a connected SIMATIC Micro Memory Card in order to operate the IM151-7 CPU interface module.
The SIMATIC Micro Memory Card is not supplied as standard with the IM151-7 CPU interface module.

Note

The IM151-7 CPU interface module goes into STOP and requests a memory reset when you remove the SIMATIC Micro Memory Card while the IM151-7 CPU interface module is in RUN state.

CAUTION

Data on a SIMATIC Micro Memory Card can be corrupted if you remove the card while it is being accessed by a write operation. You may have to delete the SIMATIC Micro Memory Card using the PG or format it in the IM151-7 CPU interface module if you remove it from the live system.
DO NOT remove the SIMATIC Micro Memory Card when the system is in RUN state; always shut down power or set the IM151-7 CPU interface module to STOP in order to prevent any write access of a programming device. If the CPU is in STOP mode and you cannot determine whether or not a programming device function is active (e.g. load / delete block), disconnect the communication lines.

WARNING

Make sure that the SIMATIC Micro Memory Card to be inserted contains the proper user program for the IM151-7 CPU interface module (for the system). The wrong user program may have fatal processing effects.

Inserting/replacing the SIMATIC Micro Memory Card

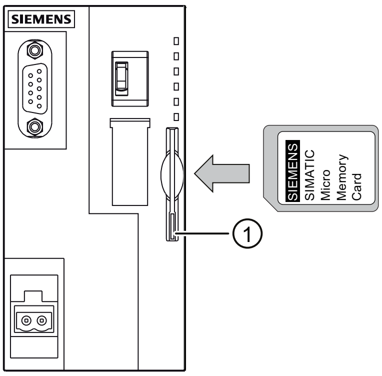
1. First of all, switch the IM151-7 CPU interface module to STOP.
2. Is there a SIMATIC Micro Memory Card inserted?

   If yes, first of all ensure that no writing PG functions (e.g. downloading blocks) are running. If you cannot ensure this, interrupt the communication connections of the IM151-7 CPU interface module.

   Press the ejector and remove the SIMATIC MMC.

   An ejector ① is located on the frame of the module receptacle to enable you to remove the SIMATIC Micro Memory Card.
   Use a small screwdriver or ball-point pen to eject.
3. Insert the ("new") SIMATIC Micro Memory Card into the receptacle so that its beveled edge points towards the ejector.
4. Gently press the SIMATIC Micro Memory Card into the IM151-7 CPU interface module until it engages.
5. Perform a reset (refer to section Resetting the IM151-7 CPU interface module using the mode selector switch (Page 93)).

IM151-7 CPU



Removing and inserting a SIMATIC Micro Memory Card

After you replace a SIMATIC Micro Memory Card in POWER OFF state, the IM151-7 CPU interface module detects

- A physically identical SIMATIC Micro Memory Card with a different content
- A new SIMATIC Micro Memory Card with the same content as the old SIMATIC Micro Memory Card

It automatically performs a Memory reset after POWER ON.

Reference

Additional information about the SIMATIC Micro Memory Card can be found in the S7-300, CPU 31xC and CPU 31x (http://support.automation.siemens.com/WW/view/en/12996906) manual, in the *Technical Data* section.

7.4.2 Initial power on

Requirements

- You have installed and wired up the ET 200S.
- The SIMATIC Micro Memory Card is inserted in the IM151-7 CPU interface module.
- The mode selector of your IM151-7 CPU interface module is set to STOP.

Initial power on of an ET 200S with IM151-7 CPU

Switch on the supply voltage for the ET 200S.

Result:

On the IM151-7 CPU interface module

- The ON LED lights up
- the STOP LED flashes at 2 Hz when the IM151-7 CPU interface module carries out an automatic memory reset
- The STOP LED lights up after the memory reset.

7.4.3 Memory reset of the IM151-7 CPU interface module using the mode selector switch

When must I perform a memory reset of the IM151-7 CPU interface module?

You must perform a memory reset of the IM151-7 CPU interface module,

- When all retentive memory bits, timers and counters have been cleared and the initial values of retentive data blocks in the load memory are to be used as actual values in the work memory.
- if the retentive memory bits, timers and counters could cause unwanted responses after "Load user program onto memory card" with the user program just downloaded to the IM151-7 CPU interface module.

  Reason: "Load user program onto memory card" does not delete the retentive areas.
- if the IM151-7 CPU interface module requests a memory reset with its STOP LED flashing at 0.5 Hz intervals. Possible reasons for this request are listed in the table below.

Table 7- 3 Possible reasons for the request to reset the memory by the IM151-7 CPU interface module

Reasons for the request to reset the memory by the IM151-7 CPU interface module	Special features
The SIMATIC Micro Memory Card has been replaced.	–
RAM error in the IM151-7 CPU interface module	–
Work memory is too small, i.e. not all the blocks of the user program on a SIMATIC Micro Memory Card can be loaded.	IM151-7 CPU with inserted SIMATIC Micro Memory Card: This causes continuous requests for a memory reset. This may be prevented by formatting the SIMATIC Micro Memory Card (see Formatting the SIMATIC Micro Memory Card (Page 96)). For additional information on the way the SIMATIC Micro Memory Card responds to a memory reset, see Memory reset and restart (Page 67).
Attempts to load faulty blocks; if a wrong instruction was programmed, for example.	

How to reset memory

There are two ways to reset the IM151-7 CPU interface module memory:

Memory reset using the mode selector switch	Memory reset using the programming device
... is described in this section.	... is only possible when the IM151-7 CPU interface module is in STOP mode (see *STEP 7 Online Help*).

Resetting the memory of the IM151-7 CPU using the mode selector switch

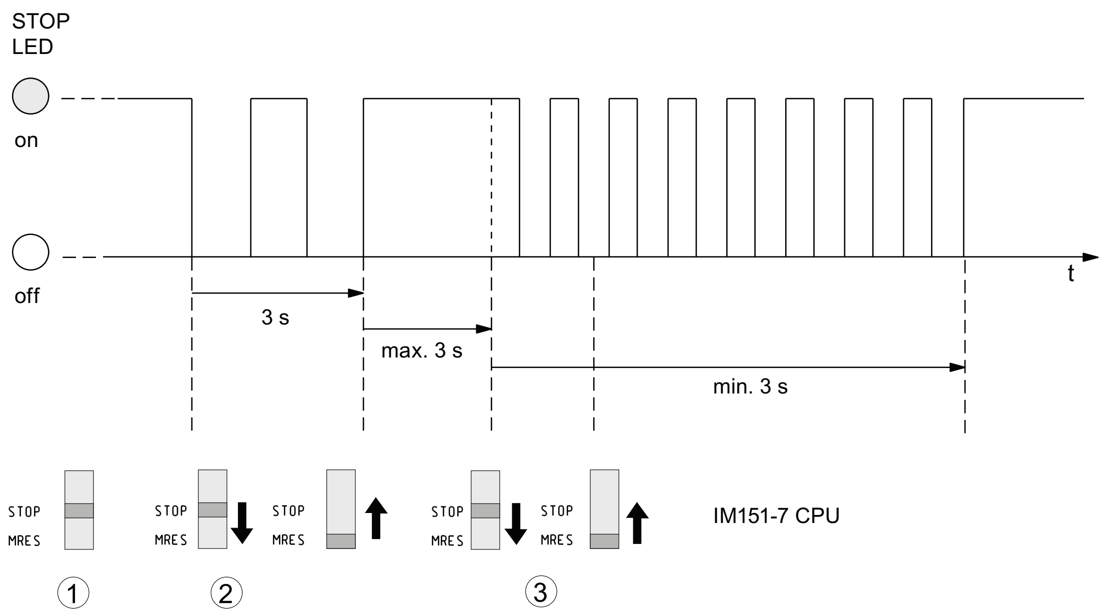
The following table contains the steps required to reset the memory of the IM151-7 CPU interface module.

Table 7- 4 Steps for performing a memory reset of the IM151-7 CPU interface module

Step	Memory reset of IM151-7 CPU
1.	Turn the key to STOP position. ①
2.	Turn the key to MRES position. Hold the key in this position until the STOP LED lights up for the second time and remains on (this takes 3 seconds). ② Now release the key.
3.	You must turn the key to MRES position again within 3 seconds and hold it there until the STOP LED flashes (at 2 Hz). ③ You can now release the switch. When the IM151-7 CPU interface module has completed the memory reset, the STOP LED stops flashing and lights up. The memory reset has been carried out by the IM151-7 CPU interface module.

The procedure described in the table above is only required if you wish to reset the IM151-7 CPU interface module memory without this being requested by the IM151-7 CPU interface module itself (STOP LED flashing slowly). If the IM151-7 CPU interface module prompts you for a memory reset, you only have to turn the mode selector briefly to the MRES position to initiate the memory reset operation.

The figure below shows how to use the mode selector switch to reset the IM151-7 CPU interface module memory:



If the IM151-7 CPU interface module prompts you for another memory reset following a successful memory reset operation, the SIMATIC Micro Memory Card may need to be reformatted (see section Formatting the SIMATIC Micro Memory Card (Page 96)).

STOP LED does not flash during the memory reset

What should I do if the STOP LED does not flash during the memory reset or if other LEDs are lit?

1. You must repeat steps 2 and 3.
2. If the IM151-7 CPU interface module does not perform a memory reset again, you must evaluate the diagnostics buffer of the IM151-7 CPU interface module.

What happens in the IM151-7 CPU interface module during a memory reset?

Table 7- 5 Internal processes in the IM151-7 CPU interface module during a memory reset

<table>
<tr><th>Event</th><th colspan="2">Action in the IM151-7 CPU interface module</th></tr>
<tr><td rowspan="4">Process in the IM151-7 CPU interface module</td><td>1.</td><td>The IM151-7 CPU interface module deletes the entire user program in the work memory.</td></tr>
<tr><td>2.</td><td>The IM151-7 CPU interface module deletes the retentive user data (flags, times, counters and DB contents).</td></tr>
<tr><td>3.</td><td>The IM151-7 CPU interface module tests its hardware.</td></tr>
<tr><td>4.</td><td>The IM151-7 CPU interface module copies the sequence-related content of the SIMATIC Micro Memory Card (load memory) to the work memory.
Tip: If the IM151-7 CPU interface module is unable to copy the content of the SIMATIC Micro Memory Card and requests a memory reset:
• Remove the SIMATIC Micro Memory Card
• Memory reset of IM151-7 CPU
• Read the diagnostics buffer</td></tr>
<tr><td>Memory contents after reset</td><td colspan="2">The user program is transferred back from the SIMATIC Micro Memory Card to the work memory and the memory utilization is indicated accordingly.</td></tr>
<tr><td rowspan="3">What's left?</td><td colspan="2">• Parameters of the MPI/DP interface:
– MPI address and highest MPI address as well as parameterized transmission rate or
– PROFIBUS address, highest PROFIBUS address, transmission rate, setting as active or passive interface when the MPI/DP interface of the IM151-7 CPU is parameterized as DP interface.</td></tr>
<tr><td colspan="2">• Data in the diagnostics buffer. (Only the last 100 entries in the diagnostics buffer are retained in the event of a POWER OFF / POWER ON.)
You can read the diagnostics buffer with the programming device (see STEP 7 Online Help).</td></tr>
<tr><td colspan="2">• The content of the operating hours counter and the time.</td></tr>
</table>

Special feature: Interface parameters of the MPI/DP interface

The table below describes which interface parameters are valid after a CPU memory reset.

Memory reset ...	MPI/DP parameters of the combined MPI/DP interface	DP parameters of the DP master interface with DP master module inserted
with inserted SIMATIC Micro Memory Card	... the MPI parameters on the SIMATIC Micro Memory Card or integrated read-only load memory are valid.	... the MPI parameters on the SIMATIC Micro Memory Card or integrated read-only load memory are valid.
	If here no parameters are stored (SDB), the previously set parameters stay valid.	If no parameters are stored here (SDB), neither are any DP interface parameters present.
without inserted SIMATIC Micro Memory Card	... are retained and valid.	... neither are any DP interface parameters present.

7.4.4 Formatting the SIMATIC Micro Memory Card

You must format the SIMATIC Micro Memory Card in the following cases

- The SIMATIC Micro Memory Card module type is not a user module.
- The SIMATIC Micro Memory Card has not been formatted.
- The SIMATIC Micro Memory Card is defective.
- The content of the SIMATIC Micro Memory Card is invalid.

  The content of the SIMATIC Micro Memory Card has been identified as invalid.
- The "Load user program" operation was interrupted as a result of POWER OFF.
- The "Write RAM to ROM" operation was interrupted as a result of POWER OFF.
- Error when evaluating the module content during Memory reset.
- Formatting error, or formatting failed.

If one of these errors has occurred, the IM151-7 CPU interface module prompts for yet another memory reset, even after the memory has already been reset. The card's content is retained until the SIMATIC Micro Memory Card is formatted, unless the "Load user program" or "Write RAM to ROM" operation was interrupted as a result of POWER OFF.

The SIMATIC Micro Memory Card is only formatted if there is a reason to do so (see above) and not, for example, when you are prompted for a memory reset after a module is changed. In this case, a switch to MRES triggers a normal memory reset for which the module content remains valid.

Use the following steps to format your SIMATIC Micro Memory Card

CAUTION
Do not use the memory cards for other than SIMATIC purposes and do not format it with third-party devices or Windows means. You overwrite the internal structure of the memory card this way and it cannot be restored. This means the memory card will become useless for SIMATIC devices.

If the IM151-7 CPU interface module is requesting a memory reset (STOP LED flashing slowly), you can format the SIMATIC Micro Memory Card with the following switch operation:

1. Toggle the switch to the MRES position and hold it there until the STOP LED lights up and remains on (after approx. 9 seconds).
2. Within the next three seconds, release the switch and toggle it once again to MRES position. The STOP LED flashes to indicate that formatting is in progress.

Note

Always perform this sequence of operation within the specified time. Otherwise, the SIMATIC Micro Memory Card will not be formatted, but rather returns to memory reset status.

See also

Memory reset of the IM151-7 CPU interface module using the mode selector switch (Page 93)

7.4.5 Resetting to the as-delivered state

Delivery state of the IM151-7 CPU interface module

In the delivery state, the properties of the IM151-7 CPU interface module are set to the following values:

Table 7- 6 Properties of the IM 151-7 CPU interface module in delivery state

Properties	Value
MPI address	2
MPI transmission rate	187.5 Kbps
Retentive memory bits, timers, counters	All retentive memory bits, timers and counters are cleared
Set retentive area for memory bits, timers and counters	Default setting (16 memory bytes, no timers, 8 counters)
Contents of the diagnostics buffer	cleared
Operating hours counter	0
Time-of-day	1.1.1994 00:00:00

Procedure

Proceed as follows to reset an IM151-7 CPU interface module to the delivery state using the mode selector:

1. Switch off the power supply.
2. Remove the SIMATIC Micro Memory Card from the receptacle (see Inserting/replacing a Micro Memory Card (Page 90)).
3. Keep the mode selector switch in the MRES position and switch the power supply on again.
4. Wait until LED lamp image 1 from the subsequent overview is displayed.
5. Release the mode selector switch, set it back to MRES within 3 seconds and hold it in this position.
6. The LED lamp image 2 from the subsequent overview is displayed.
   This lamp image lights up while the reset operation is running (approximately 5 seconds). During this period you can cancel the resetting procedure by releasing the mode selector.
7. Wait until LED lamp image 3 from the following overview is displayed and release the mode selector again.

The IM151-7 CPU interface module is reset to the delivery state. It starts without buffering (all LEDs are lit) and changes to the STOP mode.

Lamp images while resetting the IM151-7 CPU interface module

While you are resetting the IM151-7 CPU interface module to the delivery state, the LEDs light up in succession in the following lamp images:

Table 7- 7 Lamp images

LED	Color	Lamp image 1	Lamp image 2	Lamp image 3
SF	Red	□	○	Δ
BF	Red	□	□	□
ON	Green	Δ	Δ	Δ
FRCE	Yellow	○	□	□
RUN	Green	○	□	□
STOP	Yellow	○	□	□
Legend: Δ = LED lit; □ = LED dark; ○ = LED flashes at 0.5 Hz				

Note

If you remove an IM151-7 CPU interface module and re-use it in another place or place it in storage, the IM151-7 CPU should be restored to its delivery state, because the interface parameters of the MPI/DP interface are stored retentively.

7.4.6 Connecting the programming device (PG)

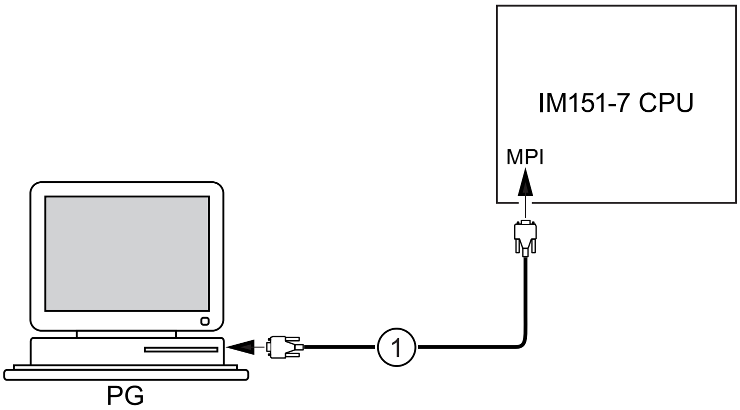
7.4.6.1 Connecting the PG to a node

Requirement

The PG must be equipped with an integrated MPI interface or an MPI card in order to connect it via MPI.

Connecting a PG to the integrated MPI interface of the IM151-7 CPU

Interconnect the PG with the MPI interface of your IM151-7 CPU by way of a prefabricated PG cable ①. You can also use a self-made connecting cable with PROFIBUS bus cable and bus connectors. The figure below illustrates the connection between the PG and the IM151-7 CPU.



Number	Designation
①	PG cable used to interconnect the PG with the IM151-7 CPU

Procedure for PROFIBUS DP

The procedure is basically the same for PROFIBUS DP, if the interface of the IM151-7 CPU interface module is set as PROFIBUS DP interface.

7.4.6.2 Connecting the PG to several nodes

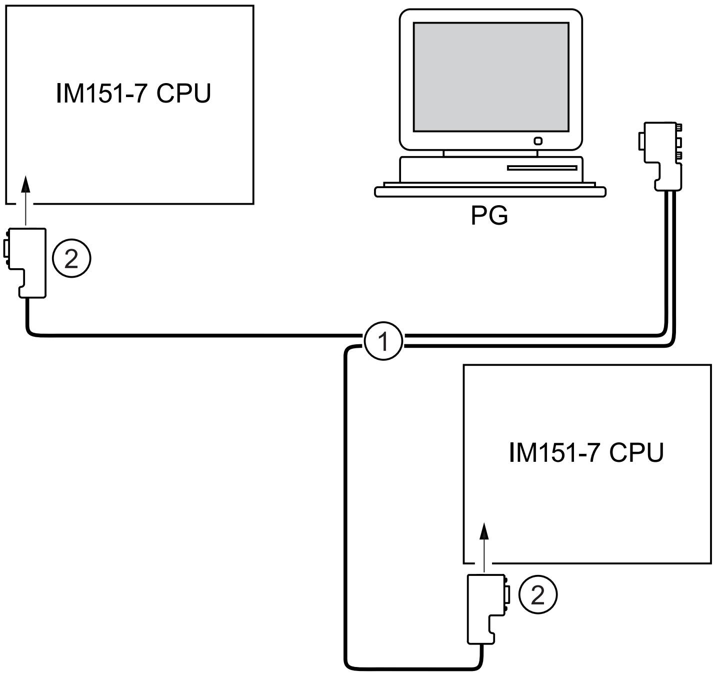
Requirement

The PG must be equipped with an integrated MPI interface or an MPI card in order to connect it to an MPI subnet.

Connecting the PG to several nodes

Use bus connectors to connect a PG which is permanently installed on the MPI subnet to the other nodes of the MPI subnet.

The following figure shows two linked IM151-7 CPUs. The two interface modules are connected via bus connectors.



Number	Designation
①	PROFIBUS bus cable
②	Connectors with enabled terminating resistors

See also

Configuring MPI and PROFIBUS subnets (Page 24)

7.4.6.3 Using the PG for commissioning or maintenance

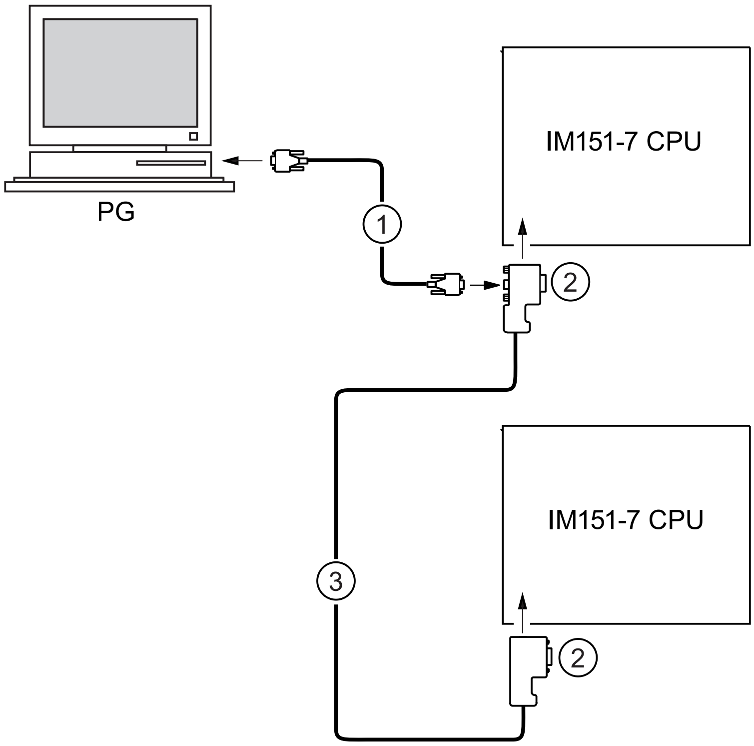
Requirement

The PG must be equipped with an integrated MPI interface or an MPI card in order to connect it to an MPI subnet.

Using the PG for commissioning or maintenance

Use a stub cable to connect the commissioning and maintenance PG to the other subnet nodes. The bus connector of this node must be equipped with a PG socket for this purpose.

The following figure below shows two linked IM151-7 CPUs to which a PG is connected.



Number	Designation
①	Stub cable for connection between programming device and IM151-7 CPU
②	Connectors with enabled terminating resistors
③	PROFIBUS bus cable for linking both IM151-7 CPUs

MPI addresses for service PGs

If there is no stationary PG, we recommend:

To connect a PG to an MPI subnet with "unknown" node addresses, set the following addresses on the service PG:

- MPI address: 0
- Highest MPI address: 126

You then determine the highest MPI address on the MPI subnet with *STEP 7* and match the highest MPI address in the PG to that of the MPI subnet.

See also

Configuring MPI and PROFIBUS subnets (Page 24)

7.4.6.4 Connecting a PG to ungrounded MPI nodes

Requirement

The PG must be equipped with an integrated MPI interface or an MPI card in order to connect it to an MPI subnet.

Connecting a PG to ungrounded nodes of an MPI subnet

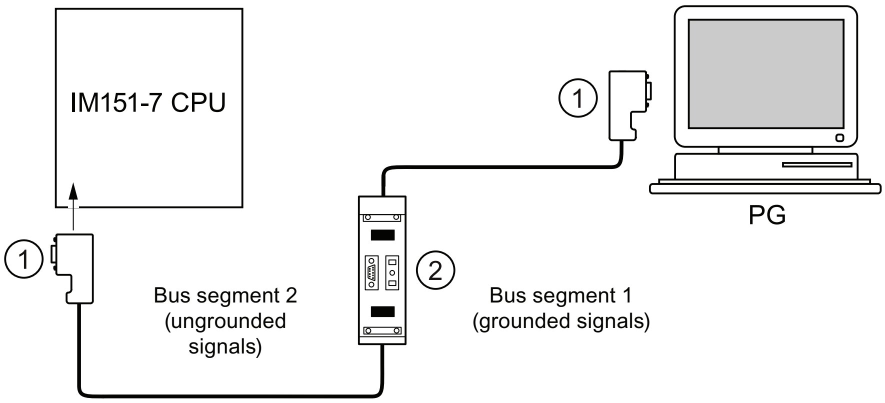
Connecting a PG to ungrounded nodes

If you configure ungrounded nodes of a subnet or an IM151-7 CPU interface module, you may only connect an ungrounded PG to the subnet or IM151-7 CPU.

Connecting a grounded PG to the MPI

You want to operate with ungrounded nodes. If the MPI at the PG is grounded, you must interconnect the nodes and the PG with an RS485 repeater. You must connect the ungrounded nodes to bus segment 2 if the PG is connected to bus segment 1 (terminals A1 B1) or to the PG/OP interface.

The following figure shows an RS 485 repeater as interface between a grounded and an ungrounded node of an MPI subnet.



Number	Designation
①	Connectors with enabled terminating resistors
②	RS485 Repeaters, with activated terminating resistors

7.4.7 Starting SIMATIC Manager

Introduction

SIMATIC Manager is a GUI for online/offline editing of S7 objects (projects, user programs, blocks, hardware stations and tools).

The SIMATIC Manager lets you

- Manage projects and libraries,
- Call *STEP 7* tools,
- Access the PLC (AS) online,
- Editing SIMATIC Micro Memory Cards.

Starting SIMATIC Manager

After installation, the SIMATIC Manager icon appears on the Windows desktop, and the Start menu contains entry SIMATIC Manager under SIMATIC.

1. Run SIMATIC Manager by double-clicking the icon, or from the Start menu (same as with all other Windows applications).

User interface

A corresponding editing tool is started up when you open the relevant objects. You start the program editor by double-clicking the program block you want to edit (object-oriented start).

Online Help

The online help for the active window is always called by pressing F1.

7.4.8 Monitoring and modifying I/Os

The "Monitor and modify variables" tool

The *STEP 7* "Monitor and modify variables" tool lets you:

- Monitor program variables in any format
- Editing (modifying) the status or content of variables in the IM151-7 CPU interface module.

Creating a variable table

You have two options of creating a variable table (VAT):

- In the LAD / FBD / STL editor by selecting the **PLC > Monitor/Modify Variables** command

  This table is also available directly online.
- In the SIMATIC Manager with the **Blocks** container open via menu item **Insert New Object > Variable table**

  This table created offline can be saved for future retrieval. You can also test it after switching to online mode.

VAT structure:

In the VAT, every address to be monitored or modified (e.g. inputs, outputs) occupies one row.

The meaning of the VAT columns is as follows:

Column text	In this field ...
Address	contains the absolute address of the variable
Icon	contains the symbolic descriptor of the variables This is identical to the specification in the Symbol Table.
Symbol comment	shows the symbol comment of the Symbol Table
Status format	contains the default format setting, e.g. HEX. You can change the format as follows: • right-click in the format field. The Format List opens. or • left-click in the format field until the relevant format appears
Status value	shows the content of the variable at the time of update
Modify value	is used to enter the new variable value (modify value)

Monitor variable

You have two options for monitoring variables:

- Updating the status values once via menu item Variable > Update Status Values

  or

- Continuous update of status values via menu item Variable > Monitor

Modifying variables

To modify variables, proceed as follows:

1. Left-click the field Modify value of the relevant variable.
2. Enter the modify value according to the data type.
3. To update modify values once, select the menu item Variable > Activate Modify Value.

   or

   Enable modify values permanently via menu item Variable > Modify.
4. In the Monitor test function, verify the modify value entry in the variable.

Is the modify value valid?

You can disable the modify value entered in the table. An invalid value is displayed same as a comment. You can re-enable the modify value.

Only valid modify values can be enabled.

Setting the trigger points

Trigger points:

- The "Trigger point for monitoring" determines the time of update for values of variables to be monitored.
- The "Trigger point for modifying" determines the time for assigning the modify values to the variables to be modified.

Trigger condition:

- The "Trigger condition for monitoring" determines whether to update values once when the trigger point is reached or continuously every time the trigger point is reached.
- The "Trigger condition for modifying" determines whether to assign modify values once or permanently to the variable to be modified.

You can customize the trigger points using the tool "Monitoring and Modifying Variables" in the menu item Variable > Set Trigger... .

Special features

- If "Trigger condition for monitoring" is set to once , the menu items Variable > Update Status Values or Variable > Monitor have the same effect, namely a single update.
- If "Trigger condition for modifying" is set to once , the menu items Variable > Update Status Values or Variable > Modify have the same effect, namely a single assignment.
- If trigger conditions are set to permanent , the said menu items have different effects as described above.
- If monitoring and modifying is set to the same trigger point, monitoring is executed first.
- If Process mode is set under Debug > Mode, values are not cyclically updated when permanent modifying is set.
  To correct or avoid error: Use the Force test function.

Saving/opening the variable table

Saving the VAT

1. After aborting or completing a test phase, you can save the variable table to memory. The name of a variable table starts with the letters VAT, followed by a number from 0 to 65535; e.g. VAT5.

Opening VAT

1. Select the menu item **Table > Open**.
2. Select the project name in the **Open** dialog.
3. In the project window below, select the relevant program and mark the **Blocks** container.
4. In the block window, select the desired table.
5. Confirm with **OK**.

Creating a connection to the IM151-7 CPU interface module

The variables of a VAT represent dynamic quantities of a user program. To monitor or modify variables, you will need to establish a connection with the relevant IM151-7 CPU interface module. Each variable table can be linked to a different IM151-7 CPU interface module.

Use the **PLC > Connect to ...** menu item to establish a connection to one of the following IM151-7 CPU interface modules:

- Configured IM151-7 CPU interface module
- Directly connected IM151-7 CPU interface module
- Available IM151-7 CPU interface module ...

The table below lists the display of variables.

Interface modules	The variables of the IM151-7 CPU interface module are displayed, ...
Configured IM151-7 CPU interface module	in S7 program (hardware station) of which the variable table is stored.
Directly connected IM151-7 CPU interface module	that is connected directly to the programming device.
Available IM151-7 CPU interface module	that is selected in the dialog window. Use the **PLC > Connect to ... > Available CPU ...** menu item to connect to an available IM151-7 CPU interface module. With this step you can create a connection to any IM151-7 CPU interface module in the network.

Controlling outputs in STOP mode of the IM151-7 CPU interface module

The function **Enable PO** resets the output disable signal for the peripheral outputs (PO). This step enables you to control the PO when the IM151-7 CPU interface module is in STOP mode.

In order to enable the POs, proceed as follows:

1. Select **Table > Open the variable table (VAT)** to open the variable table containing the POs you want to modify, or activate the window containing the corresponding VAT.
2. To control the POs of the active VAT, use the **PLC > Connect to ...** menu item to establish a connection to the required IM151-7 CPU interface module.
3. Use the **PLC > Operating Mode** menu item to open the **Operating Mode** dialog and switch the IM151-7 CPU interface module to STOP mode.
4. Enter your values in the "Modify value" column for the PO you want to modify.

   Examples:
   PO: POB 7 modify value: 2#0100 0011
   POW 2 W#16#0027
   POD 4 DW#16#0001
5. Select **Variable > Enable PO** to set "Enable PO" mode.
6. Modify the PO by selecting **Variable > Activate Modify Values**. "Enable PO" mode remains active until reset by selecting **Variable > Enable PO** once mode.

   "Enable PO" is also terminated when the connection to the programming device is dropped.
7. Return to step 4 if you want to set new values.

Note

If the IM151-7 CPU interface module changes its mode from STOP to RUN or STARTUP, for example, a message is shown.
A message is also shown if the IM151-7 CPU interface module is in RUN mode and the function "Enable PO" is selected.

7.5 Commissioning PROFIBUS DP

7.5.1 Commissioning the PROFIBUS DP network

Requirements

Requirements for commissioning a PROFIBUS DP network:

- A PROFIBUS DP network is installed.
- You have configured the PROFIBUS DP network using *STEP 7* and have assigned a PROFIBUS DP address and the address space to all the nodes.
- Note that you must also set address switches for some of the DP slaves (see the description of the relevant DP slave).
- The software as shown in the following table is required for the IM151-7 CPU interface module:

Table 7- 8 Software requirements for the IM151-7 CPU interface module

Interface module	Order number	Software required
IM151-7 CPU	6ES7151-7AA21-0AB0	as of *STEP 7* V5.2 + SP1 + HSP219 or V5.5 + SP1

DP address ranges of the IM151-7 CPU interface module

Table 7- 9 DP address ranges of the IM151-7 CPU interface module

Address range	IM151-7 CPU
Entire address range of inputs and outputs	2048 bytes
Number of those in process image for I/Os	Bytes 0 to 2047 (can be set)
	Bytes 0 to 127 (preset)

DP diagnostics addresses

DP diagnostic addresses occupy 1 byte per DP master and DP slave in the input address range. For example, at these addresses DP standard diagnostics can be called for the relevant node (LADDR parameter of SFC 13). The DP diagnostics addresses are specified in your configuration. If you do not specify any DP diagnostic addresses, *STEP 7* assigns these DP diagnostics addresses in descending order, starting at the highest byte address.

If there is an IM151-7 CPU interface module with DP master module as the master, assign two different diagnostics addresses for S7 slaves:

- Diagnostics address of the slave (address for slot 0)

  At this address all slave events are reported in the DP master (node proxy), e.g. node failure.

- Diagnostics address of the module (address for slot 2)

  Events are reported in the master (OB 82) at this address that affect the module (e.g. STOP / RUN transition of an IM 151-7 CPU as an intelligent DP slave).

See also

Commissioning the IM151-7 CPU as DP slave (Page 112)

Commissioning the IM 151-7 CPU with DP master module as a DP master (Page 119)

7.5.2 Commissioning the IM151-7 CPU as DP slave

Requirements for commissioning

- The DP master is configured and programmed.
- You must parameterize the MPI/DP interface of your IM151-7 CPU interface module as DP interface.
- Prior to commissioning, you must set the relevant parameters and configure the IM151-7 CPU as DP slave. That means that, in *STEP 7* you must
  - "switch on" the IM151-7 CPU as DP slave,
  - assign a PROFIBUS address to the IM151-7 CPU,
  - assign a slave diagnostic address to the IM151-7 CPU,
  - specify whether the DP master is an S7 DP master or another DP master,
  - specify the address ranges for data exchange with the DP master.
- All other DP slaves are programmed and configured.

Additional Information

For additional information on configuration, refer to the *STEP 7 online help*.

GSD files

If you are working on an IM 308-C or third-party system, you require a GSD file to configure the IM151-7 CPU interface module as a DP slave in a DP master system.

Note

If you wish to use the IM151-7 CPU interface module as a standard slave with the GSD file, you must not activate the "Testing, Commissioning, Routing" checkbox in the DP interface properties dialog box when you configure this slave CPU in *STEP 7*.

Configuration and parameter assignment message frame

STEP 7 assists you during configuration and parameter assignment of the DP CPU. If you need a description of the configuration and parameter assignment frame, for example in order to use a bus monitor, you can find it on the Internet (http://support.automation.siemens.com/WW/view/en/1452338).

Commissioning

Commission the IM151-7 CPU interface module as a DP slave in the PROFIBUS subnet as follows:

1. Switch on power, but keep the IM151-7 CPU interface module in STOP mode.
2. First, switch on all other DP masters/slaves.
3. Now switch the IM151-7 CPU interface module to RUN.

Startup of DP CPU as DP slave

When the IM151-7 CPU interface module is switched to RUN mode, two mutually independent operating mode transitions are executed:

- The IM151-7 CPU interface module switches from STOP to RUN mode.
- The IM151-7 CPU interface module starts data exchange with the DP master via the PROFIBUS DP interface.

Recognizing the Operating State of the DP master (Event Recognition)

The table below shows how the IM151-7 CPU interface module detects operating state transitions or data exchange interruptions as DP slave.

Table 7- 10 Detecting operating mode transitions and station failure of a DP CPU operated as DP master in the IM151-7 CPU as DP slave

Event	What happens in the DP slave?
Bus interruption (short circuit, connector removed)	• Call of OB 86 with the message **Station failure** (coming event; diagnostic address assigned to the DP slave) • With I/O access: Call of OB 122 (I/O access error)
DP master: RUN → STOP	• Call of OB 82 with the message **Module error** (incoming event; diagnostic address assigned to the DP slave; variable OB82_MDL_STOP = 1)
DP master: STOP → RUN	• Call of OB 82 with the message **Module OK** (outgoing event; diagnostic address assigned to the DP slave; variable OB82_MDL_STOP = 0)

Tip:

When commissioning the IM151-7 CPU interface module as DP slave, always program OB82 and OB86. This helps you to recognize and evaluate the respective operating states or interruptions in data exchange.

Status/modify, programming via PROFIBUS

As an alternative to the MPI interface, you can program the IM151-7 CPU interface module or execute the PG status and control functions via the PROFIBUS DP interface.

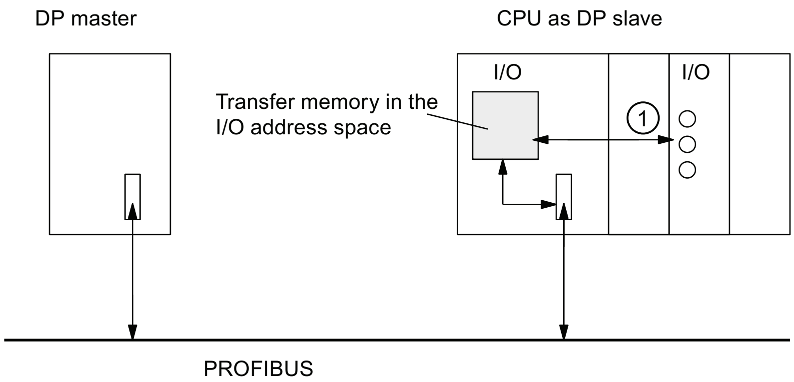
Note

The execution of status and control function via PROFIBUS DP interface extends the DP cycle.

User data transfer by way of transfer memory

The IM151-7 CPU interface module operating as intelligent DP slave provides a transfer memory to the PROFIBUS DP. User data are always exchanged between the IM151-7 CPU interface module as DP slave and the DP master by way of this transfer memory. You can configure up to 32 address ranges for this function.

This means the DP master writes its data to these transfer memory address ranges, and the IM151-7 CPU interface module reads out these data in the user program and vice versa.



Number	Description
①	The functions which control data exchange between transfer memory and the distributed I/O of the slave CPU must be implemented in the user program. The DP master cannot access this I/O directly.

Address ranges of transfer memory

In *STEP 7*, configure the I/O address ranges:

- You can configure up to 32 I/O address ranges.
- The maximum length of this address range is 32 bytes.
- You can configure a maximum of 244 input bytes and 244 output bytes.

The table below shows the principle of address ranges. You can also find this figure in the *STEP 7* configuration.

Table 7- 11 Configuration example for the address ranges of transfer memory

	Type	Master address	Type	Slave address	Length	Unit	Consistency
1	I	222	O	310	2	BYTE	Unit
2	O	0	I	13	10	Word	Total length
:							
32							
	Address ranges in the DP master CPU		Address ranges in the DP slave CPU		These paramaters of the address ranges must be the same for DP master and DP slave.		

Example program

Below, you will see a small example program for data exchange between the DP master and the DP slave. The addresses used in the example are found in the table above.

In the DP slave CPU	In the DP master CPU
`L 2 //Data preparation in the //DP slave`	
`T MB 6`	
`L IB 0`	
`T MB 7`	
`L MW 6 //Forward data to //DP master`	
`T PQW 310`	
	`L PIB 222 //continued processing of //received data in DP master`
	`T MB 50`
	`L PIB 223`
	`L B#16#3`
	`+ I`
	`T MB 51`
	`L 10 //Data preparation in the //DP master`
	`+ 3`
	`T MB 60`
	`CALL SFC 15 //Send data to the DP slave`
	`LADDR:= W#16#0`
	`RECORD:=P#M60.0 Byte20 //In the user program //of the master a block //of 20 bytes lengths //starting at MB60 //is written //consistently to the //output area //PAB0 to PAB19 //(transfer area from //master to slave)`

```
                                                        RET_VAL :=MW 22

CALL   SFC   14    //Receive data from
                   //DP master
  LADDR:=W#16#D    //In the slave, the
                   //peripheral bytes PEB13
                   //to PEB32 (data
                   //transferred from master)
                   //are read consistently and
                   //stored in MB30
                   //to MB49
  RET_VAL :=MW 20
  RECORD:=P#M30.0 byte 20
L      MB    30    //Received data
                   //continue processing
L      MB    7
+      I
T      MW    100
```

Working with transfer memory

Note the following rules when working with the transfer memory:

- Assignment of address ranges:
  - Input data of DP slaves are **always** output data of the DP master
  - Output data of DP slaves are **always** input data of the DP master
- The user can define these addresses. In the user program, access data with load/transfer instructions or with SFC 14 and SFC 15. You can also define addresses of the process image of inputs or outputs.
- The lowest address of specific address ranges is their respective start address.
- The length, unit and consistency of the address ranges for DP master and DP slave must be identical.
- The master and slave addresses may differ in logically identical transfer memory (independent logical I/O address spaces in the master and slave CPU).

Note

Assign addresses from the I/O address range of the IM151-7 CPU to the transfer memory.
You cannot use any addresses which have been assigned to transfer memory for other I/O modules.

S5 DP master

If you use an IM 308-C as DP master and the IM151-7 CPU interface module as DP slave, the following applies to the exchange of consistent data:

Use IM 308-C in the S5 control to program FB192 for enabling exchange of consistent data between the DP master and slave. With the FB192, the data of the IM151-7 CPU interface module are only output or read out in a consistent block.

S5-95 as DP master

If you use an AG S5-95 as DP master, you also have to set its bus parameters for the IM151-7 CPU interface module as DP slave.

User data transfer in STOP mode

User data is treated in transfer memory according to the STOP state of the DP master or DP slave.

- **The DP slave CPU goes into STOP**:

  Data in transfer memory of the IM151-7 CPU interface module is overwritten with "0", that is, the DP master reads "0" in direct data exchange mode.

- **The DP master goes into STOP**:

  Current data in transfer memory of the IM151-7 CPU interface module are retained and can still be read out by the IM151-7 CPU.

PROFIBUS address

For the IM151-7 CPU (DP-CPU) interface module you **must not set "126"** as a PROFIBUS address.

7.5.3 Commissioning the IM 151-7 CPU with DP master module as a DP master

Requirements for commissioning

- A DP master module is connected to the IM 151-7 CPU interface module.
- The PROFIBUS subnet has been configured.
- The DP slaves are ready for operation (see relevant DP slave manuals).
- Before commissioning, you must configure the IM151-7 CPU interface module as a DP master. That means that, in *STEP 7* you must
  - configure the IM151-7 CPU interface module as a DP master,

    ---

    Note

    You must suspend the DP master module separately as a submodule (X2) in the station window in HW Config.

  - assign a PROFIBUS address to the DP interface on the DP master module,
  - assign a master diagnostics address to the DP interface on the DP master module,
  - integrate the DP slaves into the DP master system.

    Is a DP-CPU (for example, an IM151-7 CPU interface module) a DP slave?

    If so, this DP slave will appear in the PROFIBUS DP catalog as an **already configured station.** In the DP master, assign a slave diagnostics address to this DP slave CPU. You must interconnect the DP master with the DP slave CPU and specify the address areas for data exchange with the DP slave CPU.

Additional Information

For additional information on configuration, refer to the *STEP 7 online help*.

Commissioning

Commission the IM151-7 CPU interface module with DP master module as a DP master in the PROFIBUS subnet as follows:

1. Download the PROFIBUS subnet configuration (preset configuration) created with *STEP 7* from the programming device to the IM151-7 CPU interface module.
2. Switch on all the DP slaves.
3. Switch the IM151-7 CPU interface module from STOP to RUN.

Behavior of the IM151-7 CPU interface module during commissioning

- The DP master module is installed and the IM151-7 CPU interface module is configured as a DP master
  ⇒ the IM151-7 CPU interface module switches to RUN with master functionality
- The DP master module is installed and the IM151-7 CPU interface module is not configured as a DP master
  ⇒ the IM151-7 CPU interface module switches to RUN without master functionality

Starting up the IM151-7 CPU interface module as a DP master

During startup, the IM151-7 CPU interface module checks the configured preset configuration of its DP master system against the actual configuration.

If the preset configuration = the actual configuration, the IM151-7 CPU interface module goes to RUN.

If the preset configuration ≠ the actual configuration, the configuration of parameter **Startup if preset configuration ≠ actual configuration** determines the behavior of the IM151-7 CPU interface module.

Startup when the preset configuration ≠ actual configuration = Yes (default setting)	Startup when the preset configuration ≠ actual configuration = no
IM151-7 CPU switches to RUN. (BF LED flashes if any of the DP slaves cannot be addressed).	The IM151-7 CPU remains in STOP mode, and the BF LED flashes after the set **Monitoring time for transfer of parameters to modules**. The flashing BF LED indicates that at least one DP slave cannot be addressed. In this case, check that all the DP slaves are switched on and correspond with your configuration, or read out the diagnostics buffer with *STEP 7*.

Recognizing the operating state of DP slaves (Event recognition)

The table below shows how the IM151-7 CPU interface module with DP master module acting as a DP master recognizes operating mode transitions of a CPU acting as a DP slave or any interruptions of the data exchange.

Table 7- 12 Detecting operating mode transitions and station failure of a DP CPU operated as DP slave in the IM151-7 CPU as DP master

Event	What happens in the DP master?
Bus interruption (short circuit, connector removed)	• Call of OB86 with the message **Station failure** (incoming event; diagnostics address of the DP slave assigned to the DP master) • With I/O access: Call of OB 122 (I/O access error)
DP slave: RUN → STOP	• Call of OB 82 with the message **Module error** (incoming event; diagnostics address of the DP slave assigned to the DP master; Variable OB82_MDL_STOP=1)
DP slave: STOP → RUN	• Call of OB82 with the message **Module OK** (outgoing event; diagnostics address of the DP slave assigned to the DP master; variable OB82_MDL_STOP=0)

Tip:

When commissioning the IM151-7 CPU interface module as the DP master, always program OB 82 and OB 86; they help you to recognize and evaluate data exchange errors or interruptions.

Status/modify, programming via PROFIBUS

As an alternative to the MPI interface, you can program the IM151-7 CPU interface module or execute the PG status and control functions via the PROFIBUS DP interface.

Note

The execution of status and control function via PROFIBUS DP interface extends the DP cycle.

Constant bus cycle time

This is a property of PROFIBUS DP that ensures bus cycles of exactly the same length. The "Constant bus cycle time" function ensures that the DP master always starts the DP bus cycle after a constant interval. From the perspective of the slaves, this means that they receive their data from the master at constant time intervals.

In *Step 7* HW config, you can configure constant bus cycle times for PROFIBUS subnets. A detailed description of the constant bus cycle time can be found in the *STEP 7 Online Help*.

Time synchronization

For information on time synchronization via PROFIBUS DP, please refer to the section Time synchronization (Page 44).

SYNC/FREEZE

The **SYNC** control command is used to set the DP slaves of a group to sync mode. In other words, the DP master transfers the current output data and instructs the relevant DP slaves to freeze their outputs. The DP slaves writes the output data of the next output frames to an internal buffer; the state of the outputs remains unchanged.

After each SYNC control command, the DP slaves of the selected groups transfer the output data stored in their internal buffer to the process outputs.

The outputs are only updated cyclically again after you transfer the UNSYNC control command using SFC11 "DPSYC_FR".

The **FREEZE** control command is used to set the relevant DP slaves to Freeze mode. In other words, the DP master instructs the DP slaves to freeze the current state of the inputs. It then transfers the frozen data to the input area of the IM151-7 CPU interface module.

Following each FREEZE control command, the DP slaves freeze the state of their inputs again.

The DP master does not receive the current state of the inputs cyclically once more until you have sent the UNFREEZE control command with SFC11 "DPSYC_FR".

SFC 11 is described in the *STEP 7 online help* and in the System and Standard Functions for S7-300/400 (http://support.automation.siemens.com/WW/view/en/1214574) reference manual.

Powering up the DP master system

IM151-7 CPU is a DP master
Set the power-up monitoring time for DP slaves using the **Monitoring time for parameter transfer to modules** parameter.
This means that the DP slaves must power up within the set time and must be set by the IM151-7 CPU interface module (as DP master).

PROFIBUS address of the DP master

For the IM151-7 CPU (DP-CPU) interface module you **must not set "126"** as a PROFIBUS address.

7.5.4 Direct data exchange

Requirement

In *Step 7* HW config, you can configure "Direct Data Exchange" for PROFIBUS nodes. The IM151-7 CPU interface modules with DP master module participate in a direct data exchange as receivers. If the IM151-7 CPU interface module is operated as I slave (at the combined MPI / DP interface), it can participate as sender and receiver in the direct data exchange.

Definition

"Direct data exchange" is a special communication relationship between PROFIBUS DP nodes.

Direct data exchange is characterized by the fact that the PROFIBUS DP nodes "listen" on the bus for data that a DP slave returns to its DP master. This mechanism allows the "listening node" (recipient) direct access to deltas of input data of remote DP slaves.

Address Areas

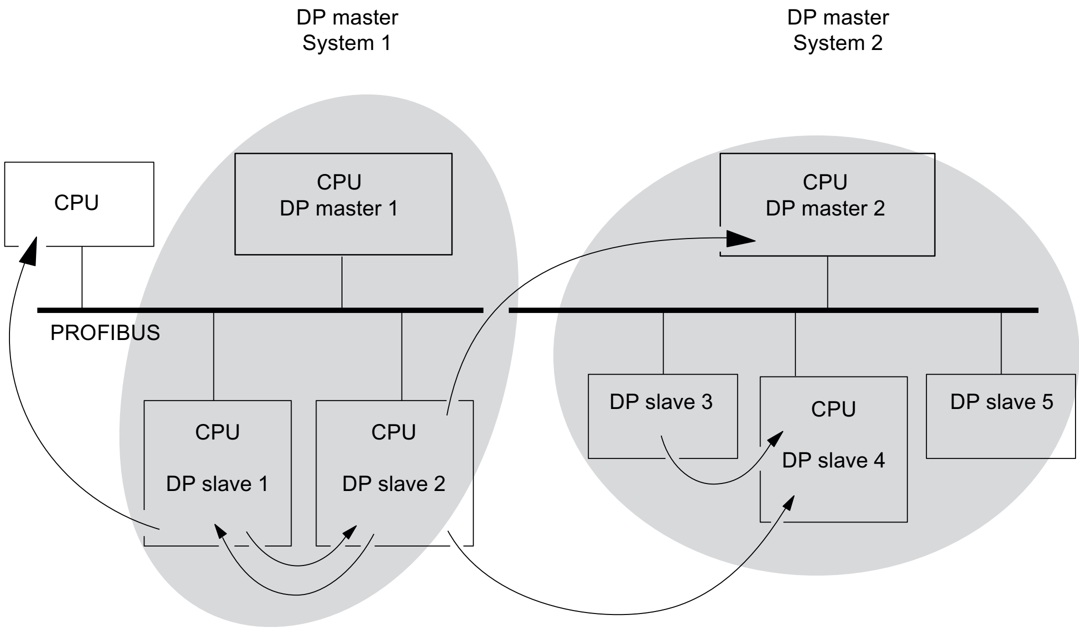
During configuration in *STEP 7*, use the I/O input addresses to specify the address area of the recipient at which the sender's data should be read.

An IM151-7 CPU interface module can be:

- Sender as DP slave
- Receiver as DP slave or DP master, or as CPU not integrated in a master system

Example: Direct data exchange via DP CPUs

The example in the figure below shows the relationships you can configure for direct data exchange. In the figure all DP masters and DP slaves (except slave 3 and slave 5) are each a DP-CPU. Note that other DP slaves (ET 200M, ET 200S, ET 200pro) can only operate as transmitters.



Service and maintenance 8

8.1 Overview

For the ET 200S with IM151-7 CPU, service and maintenance include

- Backing up the firmware to the SIMATIC Micro Memory Card
- Updating the firmware via the SIMATIC Micro Memory Card
- Updating of the firmware online
- Backing up of project data on a SIMATIC Micro Memory Card
- Replacing an IM151-7 CPU interface module
- replacing a DP master module

8.2 Backing up firmware on a SIMATIC Micro Memory Card

In which situations should I back up the firmware?

In some cases, we recommend that you back up the firmware on your IM151-7 CPU interface module.

For example, you might want to replace the IM151-7 CPU interface module in your system with an IM151-7 CPU interface module from your inventory. In this case, you should make sure that the IM151-7 CPU interface module from your inventory has the same firmware that is used in the system.

We also recommend that you create a back-up copy of the firmware for emergency situations.

SIMATIC Micro Memory Card required

You need a SIMATIC Micro Memory Card with a memory capacity ≥ 4 MB to back up the firmware.

Backing up the firmware on your IM151-7 CPU interface module to the SIMATIC Micro Memory Card

Table 8- 1 Backing up the firmware to the SIMATIC micro memory card

Step	Action required:	The following takes place in the IM151-7 CPU interface module:
1.	Insert the new SIMATIC Micro Memory Card in the IM151-7 CPU interface module.	IM151-7 CPU requests memory reset.
2.	Turn the mode selector switch to MRES position and hold it there.	-
3.	Switch supply voltage off and then on again and hold the mode selector switch in the MRES position until ...	... the STOP, RUN and FRCE LEDs start flashing.
4.	Mode selector switch to STOP.	-
5.	Mode selector switch briefly to MRES position, then let it return to STOP.	• The IM151-7 CPU interface module starts to back up the firmware on the SIMATIC Micro Memory Card. • All LEDs are lit during the backup operation. • The STOP LED flashes when the backup is complete, to indicate that the IM151-7 CPU interface module requests a memory reset.
6.	Remove the SIMATIC Micro Memory Card with the backed up firmware.	-

8.3 Updating the firmware

8.3.1 When should you update the IM151-7 CPU interface module?

After (compatible) functional expansions, or after an enhancement of operating system performance, the firmware of the IM151-7 CPU interface module should be upgraded (updated) to the latest version.

Updating an IM151-7 CPU interface module

The IM151-7 CPU interface module with order number 6ES7151-7AA20-0AB0 cannot be upgraded to the latest firmware version >= V3.3.

An IM151-7 CPU interface module with the order number 6ES7151-7AA21-0AB0 can be configured as IM151-6ES7151-7AA20-0AB0 using *STEP 7* versions < V5.5 + SP1. The new functionalities of the IM151-7 CPU V3.3 are not available with this step, if they also have to be supported by *STEP 7* (example: Encryption of blocks).

As of *STEP 7* V5.2 + SP1 you can replace even previous configurations of the IM151-7 CPU interface module together with the hardware support package HSP219 with the current configuration including a 6ES7151-7AA21-0AB0 V3.3 interface module. This step will make the full functionality available to you.

Where do I get the latest version of the firmware?

You can order the latest firmware (as *.UPD files) from your Siemens partner, or download it from the Internet (http://www.siemens.com/automation/service&support).

8.3.2 Firmware update using a SIMATIC Micro Memory Card

Table 8- 2 Firmware update using a SIMATIC Micro Memory Card

Step	Action required:	The following takes place in the IM151-7 CPU interface module:
1	**Recommendation** Before you update the firmware of your IM151-7 CPU interface module, create a backup copy of the "old" firmware on an empty SIMATIC Micro Memory Card. If problems occur during the update, you can reload your old firmware from the SIMATIC Micro Memory Card.	
2	Transfer the update files to a blank SIMATIC Micro Memory Card using *STEP 7* and your programming device. To do this, click "Update PLC / operating system" in the SIMATIC Manager. Note: You will need a SIMATIC Micro Memory Card with at least 4 MB of memory.	-
3	Switch off IM151-7 CPU power and insert a SIMATIC Micro Memory Card containing the firmware update.	-
4	Switch on power.	• The IM151-7 CPU interface module automatically detects the SIMATIC Micro Memory Card with the firmware update and runs the update. • All LEDs are lit during firmware update. • The STOP LED flashes when the firmware update complete, to indicate that the IM151-7 CPU interface module requests a memory reset.
5	Switch off IM151-7 CPU power and remove the SIMATIC Micro Memory Card containing the firmware update.	-

Result

- You have updated your IM151-7 CPU interface module with the latest firmware version.
- The address and transmission rate of the combined MPI / DP interface remain retentive. All other parameters were reset during the firmware update.

NOTICE
Aborting the firmware update by POWER ON / POWER OFF or by removal of the Micro Memory Card can lead to loss of the firmware on the CPU. In this condition only the SF-LED continues to flash at 2 Hz (all other LEDs are off). However since the boot block remains, you can regenerate the valid firmware by repeating the firmware update as described.

8.3.3 Updating the firmware online (via networks)

To update the firmware of the IM151-7 CPU interface module, you need the (*.UPD) files containing the latest firmware version.

Requirements

- An online firmware updates can be performed as of *STEP 7* V5.3.
- The interface module of the station pending a firmware update must be accessible online.
- The files containing the current firmware versions must be available in the file system of your programming device or PC. A folder may contain only the files of one firmware version.

Performing a firmware update

1. Run *STEP 7* and change to HW Config.
2. Open the station with the IM151-7 CPU interface module that is to be updated.
3. Select the IM151-7 CPU interface module.
4. Select **PLC > Update Firmware**.
5. In the **Update Firmware** dialog, select the path to the firmware update files (*.UPD) using the **Browse** button.
6. After you selected a file, the information in the lower fields of the **Update Firmware** dialog box shows you the firmware file and version for the corresponding modules.
7. Click on the **Run** button. *STEP 7* checks to determine whether the selected file can be interpreted by the IM151-7 CPU interface module and, if so, then downloads the file to the interface module. If this step requires changing the operating state of the IM151-7 CPU interface module, you will be asked to perform these tasks in the relevant dialog boxes. The IM151-7 CPU interface module then performs the firmware update independently.
8. Use *STEP 7* (read out the CPU diagnostics buffer) to verify that the IM151-7 CPU interface module can start up with the new firmware.

As an alternative you can also trigger the firmware update in SIMATIC Manager:

- Select the respective IM151-7 CPU interface module as the target CPU and then select "Update PLC / Firmware".
- Via "Available nodes", select the target CPU and then select "Update PLC / Firmware".

You can use both paths to read step 5 described above. Then continue with the remaining steps.

Result

- You have updated your IM151-7 CPU interface module online with the latest firmware version.

8.4 Backing up project data on a SIMATIC Micro Memory Card

Function principle

Using the **Save project to Memory Card** and **Fetch project from Memory Card** functions, you can save all project data to a SIMATIC Micro Memory Card, and retrieve these at a later time. For this operation, the SIMATIC Micro Memory Card can be located in an IM151-7 CPU interface module or in the SIMATIC Micro Memory Card adapter of a programming device or PC.

Project data is compressed before it is saved to a SIMATIC Micro Memory Card, and uncompressed on retrieval.

Note

In addition to project data, you may also have to store your user data on the SIMATIC Micro Memory Card. You should therefore first select a SIMATIC Micro Memory Card with sufficient free memory.

A message warns you if the memory capacity on your SIMATIC Micro Memory Card is insufficient.

The volume of project data to be saved corresponds with the size of the project's archive file.

Note

For technical reasons, you can only transfer the entire contents (user program and project data) using the **Save project to memory card** action.

Handling the functions

How you use the **Save project to memory card** / **Retrieve project from memory card** functions depends on the location of the SIMATIC micro memory card:

- When the SIMATIC MMC is inserted in the MMC slot, select a project level (for example, CPU, programs, sources or blocks) which is uniquely assigned to the IM151-7 CPU interface module from the project window in SIMATIC Manager. Select the **Target system > Save project to memory card** or **Target system > Retrieve project from memory card** menu command. All the complete project data is then written to / retrieved from the SIMATIC Micro Memory Card.
- If project data are not available on the currently used programming device (PG/PC), you can select the source CPU via "Available nodes" window. Select **PLC > Display accessible nodes** command to open the "Accessible nodes" window. Select the connection/IM151-7 CPU that contains your project data on the SIMATIC Micro Memory Card. Now select menu command **Fetch project from Memory Card**.
- If the SIMATIC MMC is located in the MMC programming unit of a PG or PC, open the "S7 memory card window" using the **File > S7 Memory Card > Open command.** Select the **Target system > Save project to memory card** or **Target system > Retrieve project from memory card** menu command. to open a dialog in which you can select the source or target project.

Note

Project data can generate high data traffic, which can lead to waiting periods of several minutes with read/write access to the IM151-7 CPU interface module especially in RUN mode.

Sample application

When you assign more than one member of your service and maintenance department to perform maintenance tasks on a SIMATIC PLC, it may prove difficult to provide quick access to current configuration data to each staff member.

If the project data of an IM151-7 CPU interface module are available locally on any IM151-7 CPU that is to be serviced, any member of the service department can access the latest project data and make any necessary changes that will then be available to all other employees.

8.5 Replacing the IM151-7 CPU interface module

Introduction

Note

If you remove an (operational) device and re-use it in another place or place it in storage, the device should be restored to its delivery state, because the bus parameters (e.g. address, transmission rate) of the combined MPI / DP interface are stored retentively.

You can replace the IM151-7 CPU interface module if it is faulty.

Requirements

To replace the IM151-7 CPU interface module you must switch off the supply voltage on the faulty IM151-7 CPU interface module .
Result: Failure of the ET 200S station and all connected components (DP slaves), if the IM151-7 CPU interface module is operated as a DP master. If the IM151-7 CPU interface module is used as I slave, the failure of the interface module is signaled as station failure to the higher-level DP master.

NOTICE
PROFIBUS DP The bus terminator function may fail if you shut down supply voltage at the first or last bus node of a bus segment.

Required tools

Screwdriver with 3 mm blade

Replacing the IM151-7 CPU interface module

The IM151-7 CPU interface module is wired, and the terminal modules are on the right:

1. Switch off the supply voltage for the faulty IM151-7 CPU interface module.
2. Remove the SIMATIC Micro Memory Card from the slot (see Inserting/replacing a SIMATIC Micro Memory Card (Page 90)).
3. If required, release the connector for the supply voltage and the RS 485 connector on the IM151-7 CPU interface module.
4. Use a screwdriver to slide the slider on the IM151-7 CPU interface module downwards until it stops. Now move the IM151-7 CPU interface module to the left.

   Note: The slider is located in the center below the IM151-7 CPU interface module.
5. Press down on the slider while swiveling the IM151-7 CPU interface module off the mounting rail.
6. Suspend the new IM151-7 CPU interface module in the mounting rail.
7. Press down on the slider while swiveling the IM151-7 CPU interface module backwards until the slider audibly engages.
8. Now move the IM151-7 CPU interface module to the right until the first terminal module.

   If a DP master module is connected:
   Move the IM151-7 CPU interface module to the right until it audibly engages on the DP master module.
9. Insert the SIMATIC Micro Memory Card removed from the defective IM151-7 CPU interface module into the slot on the new IM151-7 CPU interface module.
10. Switch on the power supply.

Behavior of the IM151-7 CPU interface module after replacement

As the SIMATIC Micro Memory Card has been changed, after the IM151-7 CPU interface module has been replaced it always automatically resets the memory and remains in STOP mode, regardless of the position of the mode selector. The IM151-7 CPU interface module can then be switched to RUN once more using the mode selector.

If the IM151-7 CPU interface module stays in STOP, you can view the cause of error in *STEP 7* (see the *STEP 7 online help*).

8.6 Replacing the DP master module

Introduction

You may replace a defective DP master module.

Requirements

To replace the DP master module you must switch off the supply voltage on the associated IM151-7 CPU interface module .
Result: Failure of the ET 200S station and all components connected to it (DP slaves).

NOTICE
PROFIBUS DP The bus terminator function may fail if you shut down supply voltage at the first or last bus node of a bus segment.

Required tools

Screw driver with 3 mm blade

Replacing the DP master module

The DP master module and the IM151-7 CPU interface module are wired, and the terminal modules are on the right:

1. Switch off the supply voltage for the respective ET 200S station (IM151-7 CPU).
2. If required, release the connector for the supply voltage and the RS 485 connector on the IM151-7 CPU interface module.
3. Use a screwdriver to slide the slider on the IM151-7 CPU interface module downwards until it stops. Now move the IM151-7 CPU interface module about 40 mm to the left.

   Note: The slider is located in the center below the IM151-7 CPU interface module.
4. Use the screwdriver to slide the slider on the DP master module downwards to the stop. Slide the faulty DP master module to the left until the connector for the backplane bus is free.

   Note: The slider is located underneath the DP master module.
5. Keeping the slider pressed down, swivel the DP master module out of the mounting rail.
6. Suspend the new DP master module in the rail and swing it downwards.
7. Slide the DP master module to the right until the first terminal module.
8. Move the IM151-7 CPU interface module to the right until it audibly engages with the DP master module.
9. If required, insert the connector for the supply voltage and the RS 485 connector once again on the IM151-7 CPU interface module.
10. Switch on the power supply.

Functions 9

9.1 Assigning parameters of the reference junction for the connection of thermocouples

Introduction

If you want to use the IM151-7 CPU interface module in an ET 200S system with thermocouples and a reference junction, set the parameters in the "Properties" section of the hardware configuration.

Parameter assignment of the reference junction

Table 9- 1 Parameter assignment of the reference junction

Parameters	Value range	Explanation
Activation of the reference junction	activated / not activated (Example, see figure below)	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 below)	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 below)	This parameter can be used to set the channel (0 / 1) for measuring the reference temperature (calculation of the compensation value) for the assigned slot of the RTD module.

RTD module parameter	Value range	Explanation
Measurement type/measurement range	Resistance / temperature measurement, e.g. • RTD-4L Pt100 standard range	If you use a channel of the RTD module for reference junction configuration, you must configure the measurement type / measurement range for this channel as **RTD-4L Pt 100 climatic range**.

TC module parameter	Value 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	This parameter allows you to enable the use of the reference junction.

Example of a parameterization dialog box

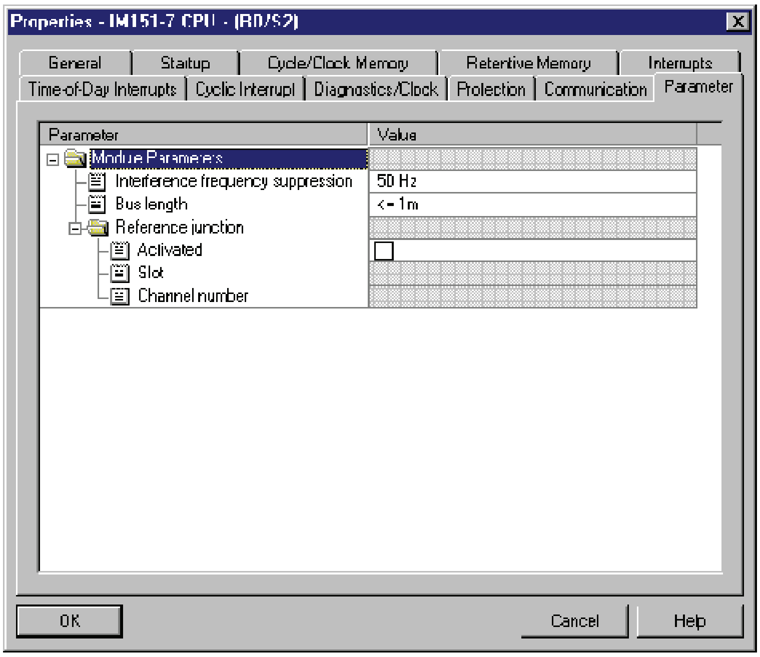


Figure 9-1 Example for a parameterization dialog box of the IM151-7 CPU module parameters in *STEP 7*

Reference

You can find detailed information on the procedure, the connection system and an example of configuration in the section entitled *Analog Electronic Modules* in the ET 200S Distributed I/O System (http://support.automation.siemens.com/WW/view/en/1144348) Operating Instructions.

9.2 Removal and insertion of modules during operation

9.2.1 Overview

The ET 200S with IM151-7 CPU supports the removal and insertion of in each case one module of the ET 200S I/O system during operation and in an energized state.

Exceptions

The IM151-7 CPU interface module 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 given here and the restrictions in the ET 200S Distributed I/O System (http://support.automation.siemens.com/WW/view/en/1144348) Operating Instructions, section *Wiring and equipping*.

> **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 modules are removed or inserted incorrectly (see ET 200S Distributed I/O System (http://support.automation.siemens.com/WW/view/en/1144348) Operating Instructions, section *Wiring and equipping*), this can cause uncontrolled system states. Adjacent modules could be affected.

9.2.2 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 diagnostics buffer entry is generated (event ID 3961_H). This takes place regardless of whether the associated power module is switched on or off.

  If the OB 83 is available on the IM151-7 CPU interface module, it 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 diagnostics buffer and the OB 122 is called up.

  If the OB 122 is available on the IM151-7 CPU interface module, it remains in RUN.

9.2.3 Procedure when modules are inserted during operation

Overview

If you insert a removed module once again in the ET 200S I/O system during operation, the IM151-7 CPU interface module initially carries out a set / actual comparison with regard to the inserted module. In doing so, the configured module is compared with the one that is actually inserted. The activities described below take place dependent on the result of the set / actual comparison.

Non-configurable modules

The following actions will take place regardless of whether the power module of the inserted module is switched on or off.

Table 9- 2 Result of the set / actual comparison in the case of non-configurable modules

Inserted module = configured module	Inserted module ≠ configured module
OB 83 is called with the corresponding diagnostics buffer entry (event -ID 3861_H).	OB 83 is called with the corresponding diagnostics buffer entry (event -ID 3863_H).
The module remains entered in the system status list as unavailable.	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 actions only take place when the power module of the inserted module is switched **on.**

Table 9- 3 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 diagnostics buffer entry (event -ID 3861_H).	OB 83 is called with the corresponding diagnostics buffer entry (event ID 3863_H).
The IM151-7 CPU interface module reconfigures the module.	The IM151-7 CPU interface module does not configure 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 actions only take place when the power module of the inserted module is switched **off.**

Table 9- 4 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 diagnostics buffer entry (event -ID 3861_H).	
When the power module is switched on, the IM151-7 CPU interface module reconfigures the module.	When the power module is switched on, the IM151-7 CPU interface module does not configure 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.

9.3 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 when assigning parameters for the power module, the diagnostics interrupt OB 82 (diagnostics address of the power module) is called with the corresponding diagnostics buffer entry (event ID 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 diagnostics interrupt OB 82 (diagnostics address of the power module) is called with the corresponding diagnostics buffer entry (event ID 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

If, during operation, you remove or insert a power module, the activities listed in section Removal and insertion of modules during operation (Page 136) take place.

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.

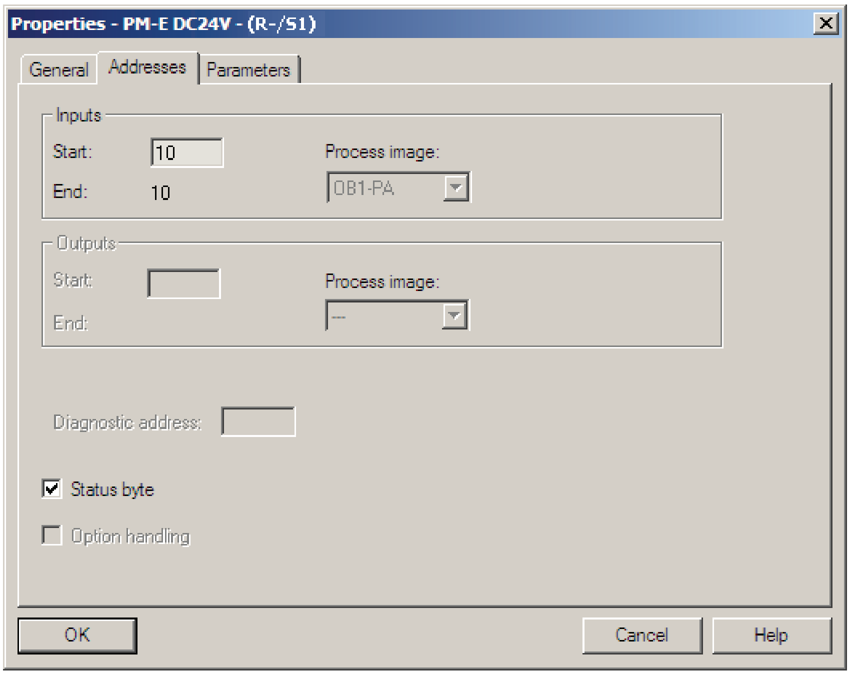
9.4 Power module with status byte

Diagnostic status of the power module

The diagnostic status of the power module can be evaluated as the input byte in the status byte.

To do this, set a check in the "Status byte" check box in the "Addresses" tab in the properties dialog of the power module.

The current status of the power module is held in a freely selectable input byte of the status byte. This is updated irrespective of whether the "No Load Voltage" diagnosis has been enabled.



Example: Cross circuit detection

An external power supply can cause a cross circuit in a power module, even when it is switched off. The status byte allows detection of the cross circuit.

Additional information

Information on the meaning of bits in the status byte can be found in the documentation for the respective power module.

Debugging functions, diagnostics and troubleshooting 10

10.1 Overview

This chapter helps you to get acquainted with tools you can use to carry out the following tasks:

- Hardware/software error diagnostics.
- Elimination of hardware/software errors.
- Testing the hardware/software – for example, during commissioning.

Note

It would go beyond the scope of this manual to provide detailed descriptions of all the tools you can use for diagnostics, testing and troubleshooting functions. Further notes are found in the relevant hardware/software manuals.

10.2 Reading/saving service data

Application

In the case of a service, for example, if the IM151-7 CPU signals the state "DEFECTIVE" (all LEDs flashing), you have the option of saving special information for analyzing the CPU state.

This information is stored in the diagnostic buffer and in the actual service data.

Select the “Target system -> Save service data” command to read and this information and save the data to a file to forward to Customer Support.

Procedure

1. If the IM151-7 CPU is in the state "DEFECTIVE" (all LEDs flashing), switch the power supply off and on (power off/on).

   Result: The IM151-7 CPU interface module is in "STOP" mode.
2. As soon as possible after the IM151-7 CPU has switched to "STOP" mode, select the respective IM151-7 CPU with the "PLC > Available nodes" menu command in the SIMATIC Manager.
3. Use the SIMATIC Manager menu command "Target system > Save service data" to save the service data.

   Result: A dialog box opens in which you specify the storage location and name of the two files.
4. Save the files.
5. Forward these files to Customer Support on request.

10.3 Identification and maintenance data of the IM151-7 CPU interface module

Definition and properties

Identification and maintenance data (I&M) are data that are stored in a module for assisting you in:

- Checking the system configuration
- Locating hardware changes in a system
- Correcting errors in a system

Identification data (I data) is information regarding the module, like for example, order number and serial number, which are partly also printed on the housing of the module. I data is manufacturer´s information about the module. It is fixed and can only be read.

Maintenance data (M data) is system-specific information, such as the installation location. M data is created during the configuration and written to the module.

I&M data enable modules to be uniquely identified online.

Reading and writing the I&M data of the IM151-7 CPU interface module with *STEP 7*

Read:

- In *STEP 7* the I&M data is displayed under "Module state – IM151-7 CPU" ("General" and "Identification" tabs) and via "Available nodes" (detailed view) (see *STEP 7 online help*).
- In the user program, the I&M data can be read via SFC 51. Specify the required SSL sublist number and the index in the input parameters of the SFC 51 (see table below).

Write:

You will always need *STEP 7* HW Config to write the M data for modules.

For example, you can enter the following data during configuration:

- Name of the automation system (device name)

  The device name is assigned when you create the station in SIMATIC Manager. In this case a "SIMATIC 300(1) station is created by default. This name can be changed at any time.
- You can enter the following data in *STEP 7* HW Config on the "General" tab under "IM151-7 CPU Properties":
  - Name of the module

    In this case, HW Config assigns a default name, e.g. IM151-7 CPU (can be changed).
  - Higher level designation of the module

    No default setting
  - Location designation of a module

    No default setting

Reading the I&M data from the IM151-7 CPU interface module with the user program

If you want to read the I&M data from the IM151-7 CPU interface module in the user program, you must read the associated system state list, specifying the relevant SSL ID and the index using SFC 51. The SSL IDs and the associated indexes are listed in the following table.

SSL sublists with I&M data

The I&M data can be found in the following SSL sublists under the specified indexes.

Table 10- 1 SSL sublists with I&M data

SSL ID W#16#...	Index W#16#...	Meaning
		Module identification
0111		an identification data record
	0001	Identification of the module This contains the module's order number and the product version.
	0006	Identification of the basic software Provides information on the software version of the module. (The IM151-7 CPU interface module has no basic software so, in this case, the identification data are the same as index 0001.)
	0007	Identification of the basic firmware Provides information on the firmware version of the module.
		Identification of a component
011C		Identification of a component
	0001	Name of the automation system The name of the automation system (device name) is saved to this parameter.
	0002	Name of the module The name of the module is saved to this parameter.
	0003	Higher level designation of the module This is a system-wide unique identifier for the module.
	000B	Location designation of a module This is the module's installation location.

For detailed information on the structure and content of the system state lists, see the System and Standard Functions for S7-300/400 (http://support.automation.siemens.com/WW/view/en/1214574) reference manual.

Additional information about reading the SSL with SFC 51 can be found in Reference Manual *System and Standard Functions for S7-300/400* or in the *Online Help for STEP 7*.

I&M data for the connected I/O devices

Information about the I&M data for the I/O devices connected to the IM151-7 CPU interface module can be found in the relevant I/O module manuals.

10.4 Debugging functions

10.4.1 Overview: Debugging functions

Determining addressed nodes with "Node flashing test"

To identify the addressed node, select **PLC > Diagnostics/Setting > Node/Flashing Test** in *STEP 7*.

A dialog appears in which you can set the flashing time and start the flashing test. The directly connected node can be identified by a flashing FRCE LED. The flashing test cannot be performed if the FORCE function is active.

Debugging functions of the software: Monitoring and modifying variables, stepping mode

STEP 7 offers you the following testing functions that you can also use for diagnostics:

- Monitoring and modifying variables

  Can be used to monitor the current values of individual variables of a user program or an IM151-7 CPU interface module on the programming device / PC. You can also assign constant values to the variables.

- Testing with program status

  You can test your program by viewing the program status of each function (result of logical links, status bit) or the data of specific registers in real-time mode.

  If you have selected the LAD programming language to be represented in *STEP 7*, the color of the symbol will indicate a closed switch or an active circuit, for example.

- Stepping mode

  When testing in single-step mode, you can process your program instructions in sequence (= single-step) and set break points. This is only possible in testing mode and not in process mode.

Note

Number of blocks and breakpoints that can be monitored with status block

With IM151-7 CPU ≥ V3.3, you can monitor two blocks at the same time and set up to four breakpoints in stepping mode.

Debugging functions of the software: Forcing variables

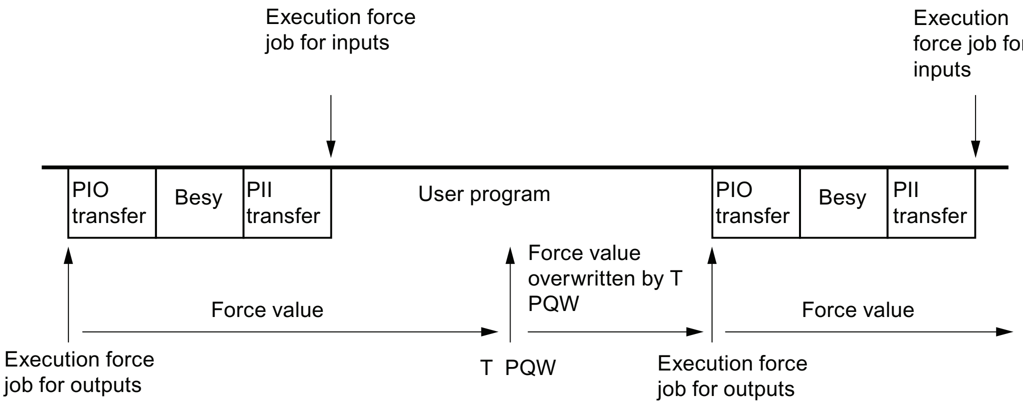
The Force function can be used to assign the variables of a user program or IM151-7 CPU interface module (also: inputs and outputs) constant values which can not be overwritten by the user program.

For example, you can use it to jumper sensors or switch outputs permanently, irrespective of the user program.

> **DANGER**
>
> This could result in severe injury or even death, and damage to property.
> Incorrect use of the Force function could result in death or severe injury, and damage to machinery or even the entire plant. Always follow the safety instructions in the *STEP 7 manuals*.

> **DANGER**
>
> **Forcing with IM151-7 CPU interface module**
>
> The force values in the process image of the **inputs** can be overwritten by write commands (such as T IB x, = I x.y, Copy with SFC, etc.) and by read I/O commands (such as L PIW x) in the user program, or by write PG/OP functions. **Outputs** initialized with forced values only return the forced value if not accessed by the user program via I/O write instructions (T PQB x, for example) or by programming device / OP write functions!
> Always ensure that forced values in the I/O process image cannot be overwritten by the user program or programming device / OP functions.

Forcing corresponds to a "cyclical control" with the IM151-7 CPU interface module



Besy: Operating system processing

Figure 10-1 Principle of forcing with IM151-7 CPU interface module

The differences between forcing and modifying variables

Table 10- 2 The differences between forcing and modifying variables

Characteristics/function	Forcing	Modifying Variables
Memory bit (M)	-	Yes
Timers and counters (T, C)	-	Yes
Data blocks (DB)	-	Yes
Inputs and outputs (I, O)	Yes	Yes
Peripheral inputs (PI)	-	-
Peripheral outputs (PO)	-	Yes
User program can overwrite modify/force values	Yes	Yes
Maximum number of force values	10	-
POWER OFF retentive	Yes	No

Reference

Details on debugging functions of the software are available in the *STEP 7 Online Help* and in the Programming with STEP 7 (http://support.automation.siemens.com/WW/view/en/18652056) manual.

For additional information on the cycle times, please refer to the "Cycle time (Page 198)" chapter.

10.4.2 Overview: Diagnostics

Introduction

Especially in the commissioning phase of a system, errors can occur. Tracking these errors might be a time-consuming effort, because errors can occur both on the hardware and software side. The many different testing functions ensure that commissioning runs smoothly.

Note

Errors during **operation** are almost always a result of faults or damage to the hardware.

Type of error

Errors that the IM151-7 CPU interface module can recognize and to which you can respond with organization blocks (OBs) can be divided into the following categories:

- Synchronous error: Errors you can relate to a specific point in the user program (error when accessing an I/O module, for example).
- Asynchronous error: Errors you can **not** relate to a specific point in the user program (cycle time exceeded, module error, for example).

Troubleshooting

Programming with foresight and, above all, knowledge and proper handling of diagnostic tools puts you into an advantageous position in error situations:

- You can reduce the effects of errors.
- It makes it easier for you to locate errors (by programming error OBs, for example).
- You can limit downtimes.

Diagnostics with LED display

The SIMATIC hardware of the distributed I/Os offers diagnostics with LEDs.

These LEDs are implemented in three colors:

LED color	Status of the IM151-7 CPU interface module
Green	Regular operation. Example: Power is on.
Yellow	Non-regular operating status. Example: Forcing is active.
Red	Fault. Example: Bus error
LED flashing	Special event Example: Memory reset

Reference

Notes on diagnosing suitable I/O modules can be found in the ET 200S Decentralized I/O System (http://support.automation.siemens.com/WW/view/en/1144348) Operating Instructions.

Diagnostics buffer

If an error occurs, the IM151-7 CPU interface module writes the cause of error to the diagnostics buffer. In *STEP 7* you use the programming device to read the diagnostics buffer. This location holds error information in plain text.

Modules with diagnostics capability that do not have their own diagnostics buffer write their error information to the diagnostic buffer of the IM151-7 CPU interface module.

When an error or an interrupt event occurs, (e.g. diagnostic interrupt for an I/O module), the IM151-7 CPU interface module switches to STOP mode, or you can respond in the user program via error / interrupt OBs. This would be OB82 in the above example.

Diagnostics with system functions

On the IM151-7 CPU interface module, we recommend that you use the more user-friendly **SFB 54** "RALRM" (called in diagnostic OB 82) to evaluate the diagnostics from centralized or decentralized I/O modules or DP slaves.

Further options for diagnostics with system functions are listed below:

- Using **SFC 51** "RDSYSST" to read an SSL sublist or an extract thereof.
- Reading the diagnostic data (slave diagnostics) of a DP slave, using **SFC 13** "DPNRM_DG"

  Every DP slave provides slave diagnostic data according to EN 50170 Volume 2, PROFIBUS. You can use SFC 13 "DPNRM_DG" to read this diagnostic data. Error information is stored in hex code. Refer to the relevant module manual for information on the meaning of the read code.

  For example, the entry of the value 50H (= dual 0101 0000) in byte 7 of the slave diagnostics for the distributed I/O module ET 200B indicates a faulty fuse or missing load voltage in channel group 2 and 3.
- Reading a data record with **SFB 52** "RDREC"

  You can use SFB 52 "RDREC" (read record) to read a specific data record from the addressed module. Data records 0 and 1 are especially suitable for reading diagnostic information from a diagnosable module.

  Data record 0 contains 4 bytes of diagnostic data that indicates the current state of a module. Data record 1 contains the 4 bytes of diagnostic data also stored in data record 0, plus module-specific diagnostic data.
- Reading the start information of the current OB using **SFC 6** "RD_SINFO"

  Information about the error can also be found in the start information of the relevant error OB.

  You can use SFC 6 "RD_SINFO" (read start information) to read the start information of the OB that was last called and not yet processed completely, and of the startup OB that was last called.
- Triggering detection of the bus topology in a DP master system with **SFC 103** "DP_TOPOL"

  The diagnostic repeater makes it easier to identify faulty modules or an interruption on the DP cable when a fault occurs during operation. The repeater acts as a slave and is able to determine the topology of a DP segment and log faults on the basis of this topology.

  You can use SFC103 "DP_TOPOL" to trigger the identification of the bus topology of a DP master system by the diagnostic repeater. SFC 103 is described in the *STEP 7 online help* and in the System and Standard Functions for S7-300/400 (http://support.automation.siemens.com/WW/view/en/1214574) reference manual. The diagnostic repeater is described in the Diagnostic Repeater for PROFIBUS DP (http://support.automation.siemens.com/WW/view/en/7915183) Manual.

10.4.3 Diagnostic functions available in STEP 7

Diagnosing with the "Diagnosing hardware" function"

Locate the cause of a module error by viewing the online information on the module. You can locate the cause of an error in the user program cycle with the help of the diagnostics buffer and of the stack content. In addition to this, you can check whether a user program is capable of running on the IM151-7 CPU interface module.

Hardware diagnostics give you an overview of the PLC status. In an overview representation, a symbol can display the error status of every module. A double-click on the faulty module opens detailed error information. The scope of this information depends on the specific module. You can view the following information:

- General information about the module (e.g. order number, version, designation) and state of the module (e.g. faulty).
- Display of module errors (channel errors, for example) in centralized I/O modules and PROFIBUS DP slaves.
- Display of messages from the diagnostics buffer.

For the IM151-7 CPU interface module, you can also view the following information about the module states:

- Cause of an error in the user program cycle.
- Indication of the cycle time (longest, shortest and last cycle).
- Performance data (number of possible inputs and outputs, memory bits, counters, timers and blocks).

For complete and current details of diagnostic functions in *STEP 7* and specific procedures, refer to the Programming with STEP 7 (http://support.automation.siemens.com/WW/view/en/18652056) manual and the STEP 7 Online Help (http://support.automation.siemens.com/WW/view/en/18652056).

10.5 Diagnostics using status and error LEDs

10.5.1 Overview

Diagnostics with LEDs is an initial tool for error localization. Usually, you evaluate the diagnostics buffer for further error localization.

The buffer contains plain text information on the error that has occurred. For example, you will find the number of the appropriate error OB here. If you generate this error OB, you can prevent the IM151-7 CPU interface module switching to STOP mode.

10.5.2 Status and error displays of the IM151-7 CPU interface module

Bus error-BF LED

The explanation of bus errors-BF LED on the IM151-7 CPU interface module or on the DP master module is available in chapters Status and error displays of the MPI/DP interface as DP slave (Page 155) or Status and error displays of the DP master module (parameterizaton as DP master) (Page 156).

Standalone operation (MPI)

The DP functionality is irrelevant in standalone operation (MPI) and no BF LED is addressed.

Table 10- 3 Status and error displays of the IM151-7P CPU interface module

LED					Meaning
SF	ON	FRCE	RUN	STOP	
Off	Off	Off	Off	Off	The IM151-7 CPU has no power supply. Remedy: Check whether the supply voltage is connected to mains and switched on.
Off	On	X	Off	On	The IM151-7 CPU is in STOP mode. To correct or avoid error: Start the IM151-7 CPU interface module.
On	On	X	Off	On	The IM151-7 CPU is in STOP mode as a result of an error. To correct or avoid error: refer to the tables below, evaluation of the SF LED
X	On	X	Off	Flashes (0.5 Hz)	The IM151-7 CPU requests memory reset.
X	On	X	Off	Flashes (2 Hz)	The IM151-7 CPU executes memory reset.
X	On	X	Flashes (2 Hz)	On	The IM151-7 CPU is in startup.
X	On	X	Flashes (0.5 Hz)	Flashes (0.5 Hz)	During transmission of blocks from the load memory to the work memory, STOP and RUN flash at 0.5 Hz until STOP mode is reached.

LED					Meaning
SF	ON	FRCE	RUN	STOP	
X	On	X	Flashes (0.5 Hz)	On	The IM151-7 CPU was paused by a programmed break point. For further information, refer to the Programming with STEP 7 (http://support.automation.siemens.com/WW/view/en/18652056) Manual.
On	On	X	X	X	Hardware or software error To correct or avoid error: refer to the tables below, evaluation of the SF LED
X	X	On	X	X	You enabled the Force function For further information, refer to the Programming with STEP 7 (http://support.automation.siemens.com/WW/view/en/18652056) Manual.
X	X	Flashes (2 Hz)	X	X	Node flashing test was activated.
Flashes	Flashes	Flashes	Flashes	Flashes	An internal system error is present in the IM151-7 CPU (all LEDs are flashing, including BF LED). The procedure is as follows: 1. Set the mode selector switch to STOP. 2. Switch the supply voltage 1L+ off and on again. 3. Read the diagnostics buffer with *STEP 7*. 4. Read the service data (see section Reading/saving service data (Page 141)). 5. Contact your local SIEMENS partner.
Flashes (2 Hz)	Off	Off	Off	Off	The IM151-7 CPU has no valid firmware; To correct or avoid error: Perform a firmware update with a Micro Memory Card (see section Firmware update using a SIMATIC Micro Memory Card (Page 128)).

X = This state is irrelevant for the current IM151-7 CPU function.

Standalone operation (MPI)

The DP functionality is irrelevant in standalone operation (MPI) and no BF LED is addressed.

Reference

- A detailed description of the OBs and SFCs / SFBs required for their evaluation can be found in the *STEP 7 online help* and in the S7-300/400 System and Standard Functions (http://support.automation.siemens.com/WW/view/en/1214574) Reference Manual.

10.5.3 Evaluating the SF LED in case of software errors

Table 10- 4 Evaluation of the SF LED (software error)

Possible error	Response of the IM151-7 CPU interface module	Possible remedies
TOD interrupt is enabled and triggered. However, a matching block is not loaded. (Software/configuration error)	Call of OB 85. The IM151-7 CPU interface module STOPs if OB 85 is not loaded.	Load OB 10 (OB number can be seen from the diagnostics buffer).
Start time of the enabled TOD interrupt was jumped, e.g. by advancing the internal clock.	Call of OB 80. The IM151-7 CPU interface module STOPs if OB 80 is not loaded.	Disable the TOD interrupt before you set the time-of-day with SFC 29.
Delay interrupt triggered by SFC 32. However, a matching block is not loaded. (Software/configuration error)	Call of OB 85. The IM151-7 CPU interface module STOPs if OB 85 is not loaded.	Load OB 20 (OB number can be seen from the diagnostics buffer).
Process interrupt is enabled and triggered. However, a matching block is not loaded. (Software/configuration error)	Call of OB 85. The IM151-7 CPU interface module STOPs if OB 85 is not loaded.	Load OB 40 (OB number can be seen from the diagnostics buffer).
Status alarm is generated, but the appropriate OB55 is not loaded.	Call of OB 85. The IM151-7 CPU interface module STOPs if OB 85 is not loaded.	Load OB55
Update alarm is generated, but the appropriate OB 56 is not loaded.	Call of OB 85. The IM151-7 CPU interface module STOPs if OB 85 is not loaded.	Load OB56
Vendor-specific alarm is generated, but the appropriate OB57 is not loaded.	Call of OB 85. The IM151-7 CPU interface module STOPs if OB 85 is not loaded.	Load OB57
Access to missing or defective I/O module when the process image is updated (software or hardware error)	Call OB 85 (depending on the parameter settings in HW Config). The IM151-7 CPU interface module STOPs if OB 85 is not loaded.	Load OB 85. The start information of the OB contains the address of the relevant I/O module. Replace the affected I/O module or eliminate the program error.
The cycle time was exceeded. Probably too many interrupt OBs called simultaneously.	Call of OB 80. The IM151-7 CPU interface module STOPs if OB 80 is not loaded. The IM151-7 CPU interface module switches to STOP even though OB 80 is loaded if twice the cycle time was exceeded without the cycle time being triggered again.	Extending the cycle time (*STEP 7* – Hardware configuration), changing the program structure. To correct or avoid error: If necessary, retrigger cycle time monitoring by calling SFC 43

Possible error	Response of the IM151-7 CPU interface module	Possible remedies
Programming error • Block not loaded • Wrong block number • Wrong timer/counter number • Read/write access to wrong area • etc.	Call of OB 121. The IM151-7 CPU interface module STOPs if OB 121 is not loaded.	Eliminate the programming error. The *STEP 7* testing function helps you to locate the error.
I/O access errors An error has occurred when I/O module data was accessed	Call of OB 122. The IM151-7 CPU interface module STOPs if OB 122 is not loaded.	Check I/O module addressing in HW Config or whether an I/O module/DP slave has failed.

Tip:

- You can use SFC 39 to disable all interrupts and asynchronous error events.

Note

The shorter the selected cyclic interrupt period, the more likely it is that cyclic interrupt errors will occur. You must take into account the operating system times of the IM151-7 CPU interface module, the user program runtime and extension of the cycle time by active programming device functions, for example.

Reference

A detailed description of the OBs and on SFCs required for their evaluation can be found in the *STEP 7 Online Help* and in the S7-300/400 System and Standard Functions (http://support.automation.siemens.com/WW/view/en/1214574) reference manual.

10.5.4 Evaluating the SF LED in case of hardware errors

Table 10- 5 Evaluating the SF LED (hardware error)

Possible error	Response of the IM151-7 CPU interface module	Possible remedies
A centralized I/O module was removed or inserted while the system was in RUN mode.	Call of OB 83. The IM151-7 CPU interface module STOPs if OB 83 is not loaded. If more than one module is removed, the IM151-7 CPU interface module always switches to STOP.	Load OB 83.
A distributed module was removed from or inserted on the PROFIBUS DP while the system was in RUN mode.	Call of OB 86. The IM151-7 CPU interface module STOPs if OB 86 is not loaded. If the module was integrated using a GSD file: Call of OB 82. The IM151-7 CPU interface module STOPs if OB 82 is not loaded.	Load OB86 or OB82.
An I/O module with diagnostic capability reports a diagnostic interrupt.	Call of OB 82. The IM151-7 CPU interface module STOPs if OB 82 is not loaded.	Response to the diagnostic event, which depends on the parameter assignments for the I/O module.
Attempt to access a missing or faulty I/O module. Loose connector (software or hardware error).	Call of OB 85, if access was attempted during update of the process image (OB 85 call must be enabled accordingly in the parameters). Call OB 122 for direct I/O access. The IM151-7 CPU interface module STOPs if the OB is not loaded.	Load OB 85 or OB 122. The start information of the OB contains the address of the relevant I/O module. Replace the relevant I/O module, fix the connector or eliminate the program error.
SIMATIC Micro Memory Card is faulty.	The IM151-7 CPU interface module switches to STOP and requests a memory reset.	Replace the SIMATIC Micro Memory Card, reset the IM151-7 CPU interface module memory, transfer the program again, then set the IM151-7 CPU interface module to RUN mode.

Reference

A detailed description of the OBs and on SFCs required for their evaluation can be found in the *STEP 7 Online Help* and in the S7-300/400 System and Standard Functions (http://support.automation.siemens.com/WW/view/en/1214574) reference manual.

10.5.5 Status and error displays of the MPI/DP interface as DP slave

Explanation of the BF LED at the IM151-7 CPU interface module

Table 10- 6 BF LED

IM151-7 CPU		
SF	**BF**	**Meaning**
On	On/flashes	PROFIBUS DP interface error. To correct or avoid error: See the tables below

Table 10- 7 BF LED at the IM151-7 CPU interface module lights up

Possible error	Response of the IM151-7 CPU interface module	Possible remedies
• Bus fault (hardware fault) • DP interface error • I slave operation: No connection to the DP master – IM151-7 CPU is active bus node ⇒ bus short circuit – IM151-7 CPU is passive bus node ⇒ transmission rate search: no active node on bus, DP master not present or turned off or bus connection interrupted	Call of OB 86, if the IM151-7 CPU is in RUN mode and communication between the DP master and DP slave functioned properly before the error occurred. The IM151-7 CPU interface module STOPs if OB 86 is not loaded.	• Check to see if the connector for PROFIBUS DP is inserted properly. • Check the bus cable for a short-circuit or break. • Analyze the diagnostic data. Edit the configuration.

Table 10- 8 BF LED at the IM151-7 CPU interface module flashes

Possible error	Response of the IM151-7 CPU interface module	Possible remedies
Bus error during operation as I slave (active bus node) Possible causes: • I slave is configured incorrectly • Incorrect, but permitted station address configured • configured address ranges of the actual structure do not correspond with the set structure • Station failure of a parameterized sender in direct data exchange • DP master not available or switched off	Call of OB 86, if IM151-7 CPU is in RUN mode and communicated as a DP slave with the DP master before the error occurred. The IM151-7 CPU interface module STOPs if OB 86 is not loaded.	• Check the IM151-7 CPU interface module. • Verify that the bus connector is properly seated. • Check for breaks in the bus cable to the DP master. • Check the configuration and parameterization of the parameterized address ranges for the DP master.

Reference

A detailed description of the OBs and on SFCs required for their evaluation can be found in the *STEP 7 Online Help* and in the S7-300/400 System and Standard Functions (http://support.automation.siemens.com/WW/view/en/1214574) reference manual.

10.5.6 Status and error displays of the DP master module (parameterizaton as DP master)

Explanation of the BF LED on the DP master module

Table 10- 9 BF LED

IM151-7 CPU	DP master module	Meaning
SF	BF	
On	On/flashes	PROFIBUS DP interface error. To correct or avoid error: See the tables below

Table 10- 10 BF LED on the DP master module lights up

Possible error	Response of the IM151-7 CPU interface module	Possible remedies
• Bus fault (hardware fault) • Short-circuit on the bus	Call of OB 86 (if the IM151-7 CPU interface module is in RUN and has previously run DP slaves that are now failing). The IM151-7 CPU interface module STOPs if OB 86 is not loaded.	• Check to see if the connector for PROFIBUS DP is inserted properly. • Check the bus cable for short-circuit. • Analyze the diagnostic data. Edit the configuration.

Table 10- 11 BF LED on the DP master module flashes

Possible error	Response of the IM151-7 CPU interface module	Possible remedies
The IM151-7 CPU interface module is a DP master: • Failure of a connected station • At least one of the configured slaves cannot be accessed. • Incorrect configuration (configured address areas of the actual structure do not correspond with the set structure.)	Call of OB 86 (if the IM151-7 CPU interface module is in RUN and connected DP slaves have since failed). The IM151-7 CPU interface module STOPs if OB 86 is not loaded.	• Check whether the bus cable on the DP master module is connected to the IM151-7 CPU interface module and that the bus is not interrupted. • Wait until the IM151-7 CPU interface module has started up. If the LED does not stop flashing, check the DP slaves or evaluate the diagnostic data for the DP slaves. • Check the settings for the configured address areas for the DP master.

Reference

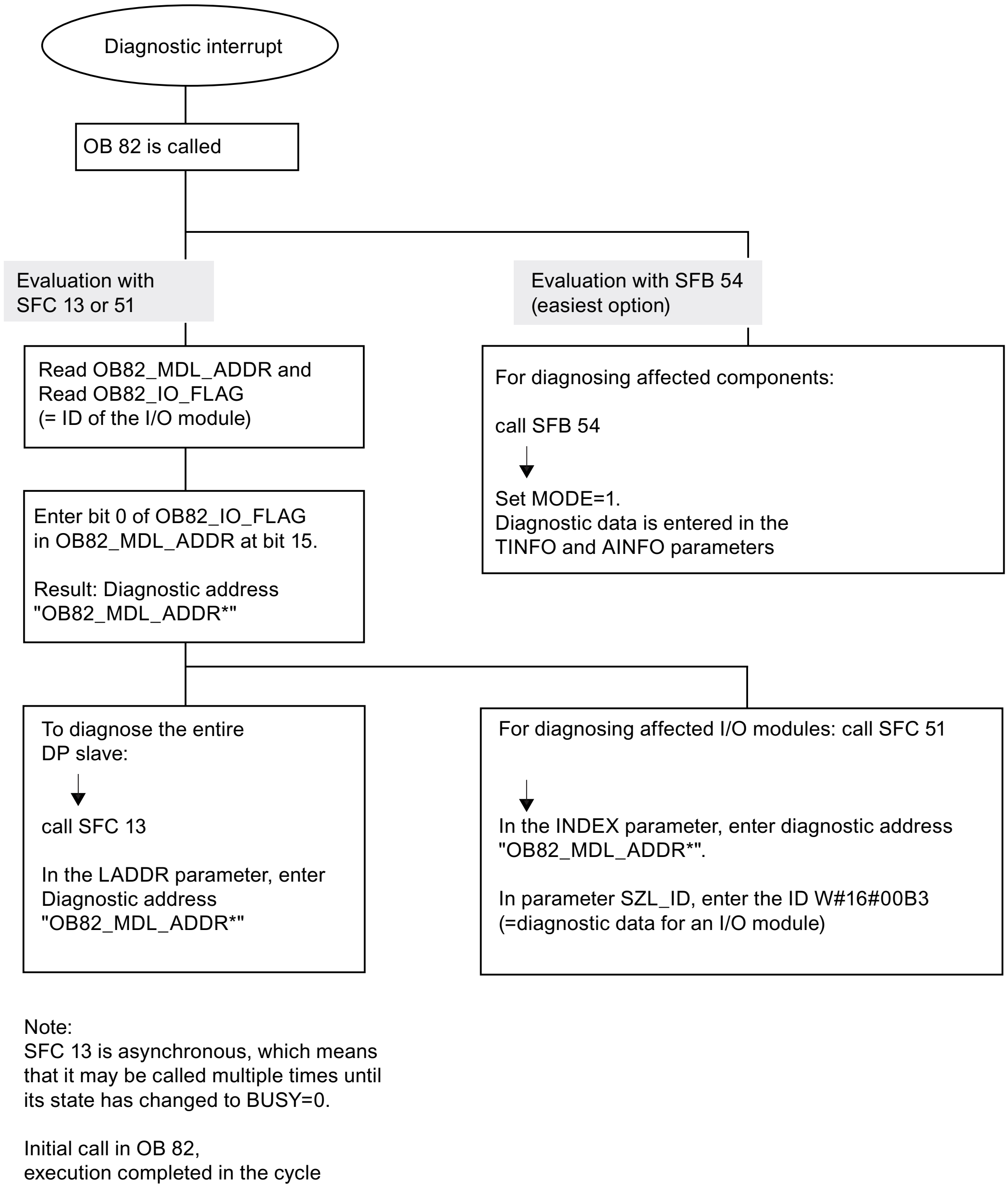
A detailed description of the OBs and on SFCs required for their evaluation can be found in the *STEP 7 Online Help* and in the S7-300/400 System and Standard Functions (http://support.automation.siemens.com/WW/view/en/1214574) reference manual.

10.6 Diagnostics on the PROFIBUS DP

10.6.1 Diagnostics of the IM151-7 CPU interface module as a DP master

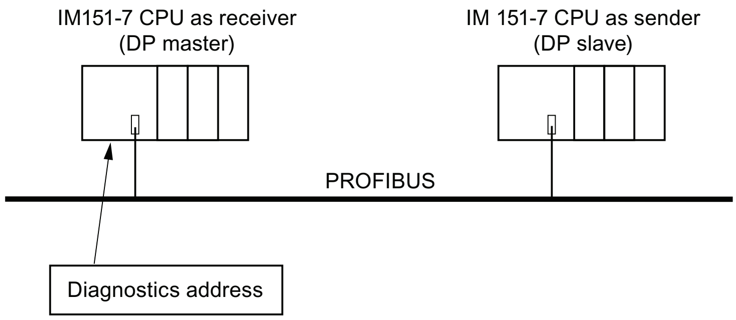
Evaluate diagnostics in the user program

The figure below illustrates the procedure for evaluating diagnostics data in the user program.



Diagnostics addresses for DP masters and DP slaves

With the IM151-7 CPU interface module you assign diagnostics addresses for the PROFIBUS DP. Verify in your configuration that the DP diagnostics addresses are assigned once to the DP master (for the DP master module) and once to the DP slave.



Description of the DP master configuration	Description of the DP slave configuration
When you configure the DP master, assign two different diagnostics addresses for an intelligent DP slave, that is, one diagnostics address for slot 0, and one for slot 2. These two addresses perform the following functions: • The diagnostics address for slot 0 reports in the master all events relating to the entire slave (station representative), for example, node failure. • The diagnostics address for slot 2 is used to report events concerning this slot. For example, if the IM151-7 CPU is acting as an intelligent DP slave, it returns the diagnostic interrupts for operating state transitions. These diagnostics addresses are referred to as *assigned to the DP master* below. These diagnostics addresses are used by the DP master to obtain information about the state of the DP slave, or about bus interruptions.	When you configure the DP slave, you also assign it a diagnostics address (in the associated DP slave project). This diagnostics address is referred to as *assigned to the DP slave* below. This diagnostics address is used by the DP slave to obtain information about the state of the DP master, or bus interruptions.

Event recognition

The table below shows how the IM151-7 CPU interface module acting as a DP master recognizes operating mode transitions of a CPU acting as a DP slave or any interruptions of the data exchange.

Table 10- 12 Event recognition of the IM151-7 CPU interface module as a DP master

Event	What happens in the DP master?
Bus interruption (short circuit, connector removed)	• Call of OB 86 with the message **Station failure** (incoming event; diagnostics address of Slot 0 of the DP slave that is assigned to the DP master) • With I/O access: call of OB 122 (I/O access error)
DP slave: RUN → STOP	• Call of OB 82 with the message **Module error** (incoming event; diagnostics address of Slot 2 of the DP slave that is assigned to the DP master; Variable OB82_MDL_STOP=1)
DP slave: STOP → RUN	• Call of OB 82 with the message **Module OK.** (outgoing event; diagnostics address of Slot 2 of the DP slave that is assigned to the DP master; Variable OB82_MDL_STOP=0)

Evaluation in the user program

The table below shows how you can, for example, evaluate RUN to STOP transitions of the DP slave in the DP master.

Table 10- 13 Evaluating RUN to STOP transitions of the DP slave in the DP master

In the DP master	In the DP slave (e.g. CPU 31x-2 DP)
Diagnostics addresses: (Example) Master diagnostics address =1023 Slave diagnostics address =1022 (Slot 0 of slave) (Diagnostic) address for "Slot 2"=1021 (Slot 2 of slave)	Diagnostics addresses: (Example) Slave diagnostics address =422 Master diagnostics address = irrelevant
The IM151-7 CPU interface module calls OB 82 with the following information: • OB82_MDL_ADDR:=1021 • OB82_EV_CLASS:=B#16#39 (incoming event) • OB82_MDL_DEFECT:=module fault Tip: This information is also available in the diagnostics buffer of the IM151-7 CPU interface module. In the user program you should also include SFC 13 "DPNRM_DG" for reading DP slave diagnostic data.	CPU: RUN -> STOP The CPU generates a DP slave diagnostic message frame

10.6.2 Reading out slave diagnostic data

The slave diagnostic data is compliant with EN 50170, Volume 2, PROFIBUS. Depending on the DP master, diagnostic data for all DP slaves conforming to standard can be read with *STEP 7*.

Diagnostics addresses for the receiving station with direct data exchange

For direct data exchange, you assign a diagnostics address in the receiving station:

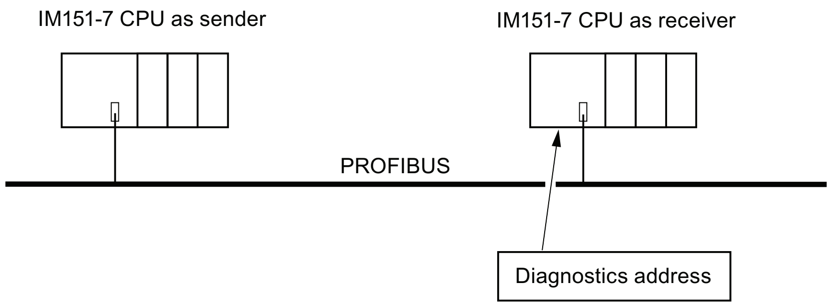


Figure 10-2 PROFIBUS DP diagnostics address

In this figure, you see that assign a diagnostics address to the receiving station in your configuration. The receiving station receives information about the status of the transmitting station or about a bus interruption by means of this diagnostics address.

Reading out the diagnostic data

The table below shows you how the various DP master systems can read diagnostic information from a slave.

Table 10- 14 Reading out diagnostic data in the master system, using *STEP 5* and *STEP 7*

Automation system with DP master	Block or tab in *STEP 7*	Application	Additional Information
SIMATIC S7/M7	"DP Slave Diagnostics" tab	Display the slave diagnostics as plain text on the *STEP 7* user interface	Refer to the keyword *Hardware diagnostics* in the *STEP 7 online help* and in the Programming with STEP 7 (http://support.automation.siemens.com/WW/view/en/18652056) Manual
	SFB 54 "RALRM"	Reading additional interrupt information from a DP slave or local module from the relevant OB.	System and Standard Functions Reference Manual (http://support.automation.siemens.com/WW/view/en/1214574)
	SFC13 "DP NRM_DG"	Reading out slave diagnostics (store in the data area of the user program)	System and Standard Functions *Reference Manual*
	SFC 51 "RDSYSST"	Reading SSL sublists. In the diagnostic interrupt, call SFC 51 with the SSL ID W#16#00B4 and then read out the SSL of the slave CPU.	System and Standard Functions *Reference Manual*
	SFB 52 "RDREC" and SFC 59 "RD_REC"	Reading the data records of S7 diagnostics (stored in the data area of the user program)	System and Standard Functions *Reference Manual*
	FB 125/FC 125	Evaluating slave diagnostic data	On the Internet (http://support.automation.siemens.com/WW/view/en/387257)
SIMATIC S5 with IM 308-C operating in DP master mode	FB 192 "IM308C"	Reading slave diagnostic data (store in the data area of the user program)	Distributed I/O System ET 200 Manual (http://support.automation.siemens.com/WW/view/en/21667381)

Example for reading the slave diagnosis with FB 192 "IM308C"

This shows you an example of how to use FB 192 in the **STEP 5** user program to read out slave diagnostics data for a DP slave.

Assumptions regarding the STEP 5 user program

For this **STEP 5** user program it is assumed that:

- The IM 308-C operating in DP master mode uses the page frames 0 to 15 (number 0 of IM 308-C).
- The DP slave is assigned PROFIBUS address 3.
- Slave diagnostics data should be stored in DB 20. You may also use any other DB.
- Slave diagnostic data consists of 26 bytes.

STEP 5 user program

STL			Explanation
	:A	DB 30	
	:SPA	FB 192	
Name	:IM308C		
DPAD	:	KH F800	//Default address range of IM 308-C
IMST	:	KY 0.3	//IM no. = 0, PROFIBUS address of the DP slave = 3
FCT	:	KC SD	//function: Read slave diagnostics
GCGR	:	KM 0	//not evaluated
TYP	:	KY 0, 20	//S5 data area: DB 20
STAD	:	KF +1	//Diagnostic data starting at data word 1
LENG	:	KF 26	//Length of diagnostic data = 26 bytes
ERR	:	DW 0	//Error code storage in DW 0 of DB 30

Example of reading out S7 diagnostic data with SFC 59 "RD REC"

Here you will find an example of how to use SFC 59 to read S7 diagnostics data records for a DP slave in the STEP 7 user program. The process of reading the slave diagnostics is similar for SFC 13.

Assumptions for the *STEP 7* user program

For this *STEP 7* user program it is assumed that:

- Diagnostic data for the input module at address 200_H is to be read.
- Data record 1 is to be read out.
- Data record 1 is to be stored in DB 10.

STEP 7 user program

STL	Explanation
CALL SFC 59	
REQ :=TRUE	//Request to read
IOID :=B#16#54	//Identifier of the address range, here the I/O input
	//Logical address of the module
LADDR :=W#16#200	//Data record 1 is to be read
RECNUM :=B#16#1	//An error code is output if an error occurs
RET_VAL :=MW2	//Read operation not finished
BUSY :=MO.0	//DB 10 is target area for the read data record 1
RECORD :=P# DB10.DBX 0.0 BYTE 240	

Note:

Data is only returned to the target area if BUSY is reset to 0 and if no negative RET_VAL has occurred.

Diagnostics addresses

With the IM151-7 CPU interface module you assign diagnostics addresses for the PROFIBUS DP. When configuring the project, make sure that the DP diagnostics addresses are assigned once to the DP master and once to the DP slave.

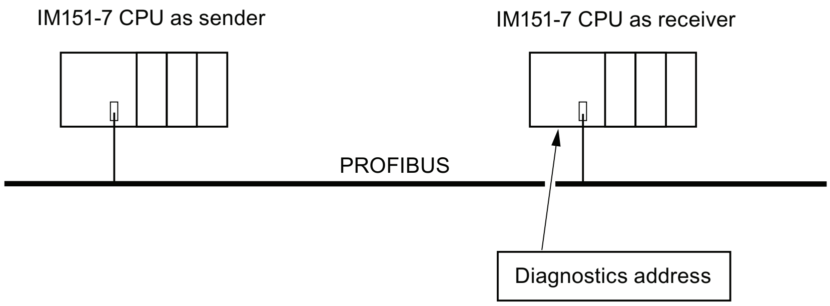


Figure 10-3 PROFIBUS DP diagnostics address

Description of the DP master configuration	Description of the DP slave configuration
When you configure the DP master, assign two different diagnostics addresses for an intelligent slave, that is, one diagnostics address for slot 0, and one for slot 2. Functions of those two addresses: • The diagnostics address for slot 0 is used to report in the master all events relating to the entire slave (station substitute), for example, node failure. • The diagnostics address for slot 2 is used to report events concerning this slot. For example, if the CPU is acting as an I-slave, it returns the diagnostic interrupts for operating mode transitions. From now on, these diagnostics addresses is referred to as *assigned to the DP master* . These diagnostics addresses are used by the DP master to obtain information about the status of the DP slave or about bus interruptions.	When you configure the DP slave, you also assign it a diagnostics address (in the associated DP slave project). Below, this diagnostics address is referred to as *assigned to DP slave*. This diagnostics address is used by the DP slave to obtain information about the state of the DP master, or bus interruptions.

Event recognition

The table below shows how the IM151-7 CPU interface module detects operating state transitions or data exchange interruptions as DP slave.

Table 10- 15 Event recognition of the IM151-7 CPU interface module as a DP slave

Event	What happens in the DP slave?
Bus interruption (short circuit, connector removed)	• Calls OB86 with the message **Station failure** (incoming event; diagnostics address of the DP slave, assigned to the DP slave) • With I/O access: call of OB 122 (I/O access error)
DP master: RUN → STOP	• Calls OB82 with the message **Faulty module** (incoming event; diagnostics address of the DP slave assigned to the DP slave; variable OB82_MDL_STOP = 1)
DP master: STOP → RUN	• Call of OB82 with the message **Module OK.** (outgoing event; diagnostics address assigned the DP slave; variable OB82_MDL_STOP = 0)

Evaluation in the user program

The table below shows how you can, for example, evaluate RUN-STOP transitions of the DP master in the DP slave (see also the previous table).

Table 10- 16 Evaluating RUN-STOP transitions in the DP Master/DP Slave

In the DP master	In the DP slave
Diagnostics addresses: (Example) Master diagnostics address = **1023** Slave diagnostics address in the master system = **1022** (Slot 0 of slave) (Diagnostic) address for "Slot 2" = **1021** (Slot 2 of slave)	Diagnostics addresses: (Example) Slave diagnostics address = **422** Master diagnostics address = irrelevant
CPU: RUN → STOP	→ The CPU calls OB 82 with the following information, for example: • OB 82_MDL_ADDR: = 422 • OB82_EV_CLASS: = B#16#39 (incoming event) • OB82_MDL_DEFECT: = Module fault Tip: The CPU diagnostics buffer also contains this information.

10.6.3 Interrupts on the DP Master

Interrupts with S7 DP master

Process interrupts from an intelligent slave with SFC 7

In the IM151-7 CPU interface module as DP slave, you can trigger a process interrupt in the DP master from the user program.

The call of SFC 7 "DP_PRAL" triggers a call of OB 40 in the user program of the DP master. The SFC 7 allows you to forward interrupt information to the DP master in a double word. This information can then be evaluated in the OB40_POINT_ADDR variable in the OB40. The interrupt information can be programmed user-specific. You will find a detailed description of SFC7 "DP_PRAL" in the System and Standard Functions for S7-300/400 (http://support.automation.siemens.com/WW/view/en/1214574) Reference Manual.

Setting user-defined interrupts of Intelligent Slaves using SFB 75

In the IM151-7 CPU interface module as DP slave, you can trigger a any interrupt in the DP master from the user program. SFB 75 "SALRM" is used to send a process or diagnostic interrupt from a slot in the transfer area (virtual slot) to the associated DP master from the user program on an intelligent slave. This starts the associated OB on the DP master.

Additional interrupt-specific information may be included. You can read this additional information in the DP master using SFB 54 "RALRM".

Interrupts with another DP master

If you operate the IM151-7 CPU interface module with another DP master, these interrupts are recreated in the device-specific diagnostic data of the IM151-7 CPU interface module. You have to post-process the relevant diagnostic events in the user program of the DP master.

Note

In order to allow the evaluation of diagnostics and process interrupts by means of device-specific diagnostics using a different DP master, please note that:

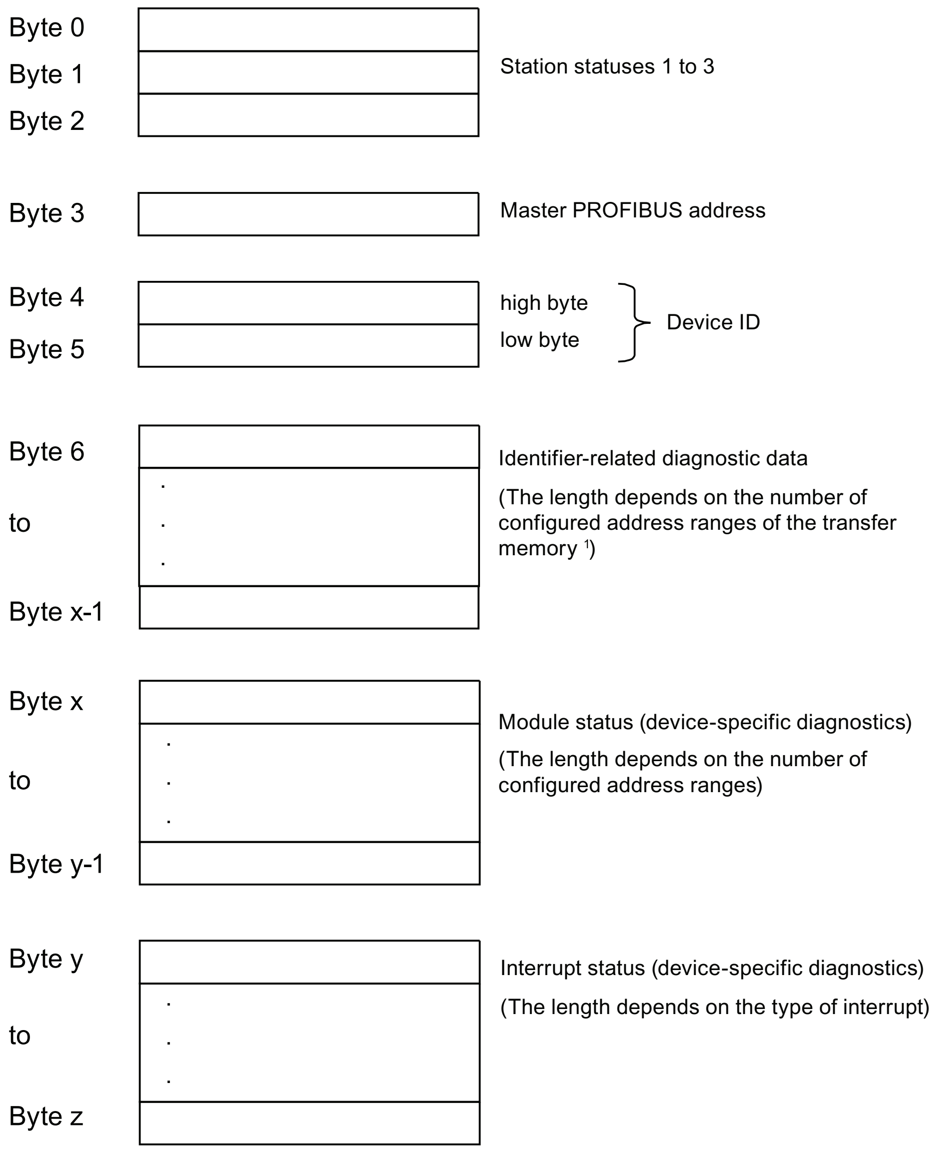
The DP master should be able to save the diagnostics messages, which means the diagnostics messages should be stored in a ring buffer within the DP master. For example, if the DP master can not save the diagnostic messages, only the last incoming diagnostic message would be saved.

In your user program, you have to poll the relevant bits of the device-specific diagnostics data in cyclic intervals. Make allowances for the PROFIBUS DP bus cycle time, for example, to be able to poll these bits at least once and in synchronism to the bus cycle time.

You cannot utilize process interrupts in device-specific diagnostics with an IM 308-C as a DP master, because only incoming events but not outgoing events are reported.

10.6.4 Structure of the slave diagnostics when the CPU is operated as I-slave

Syntax of the diagnostics datagram for slave diagnostics



[1] Exception: In case of an incorrect configuration from the DP master, the DP slave interprets 35 configured address ranges (46_H in byte 6)

Figure 10-4 Structure of slave diagnostic data

Station Status 1

Table 10- 17 Structure of station status 1 (Byte 0)

Bit	Meaning	Remedy
0	**1:** DP slave cannot be addressed by DP master.	• Is the correct DP address set on the DP slave? • Are the bus connectors connected? • Does the DP slave have power? • Correct configuration of the RS485 Repeater? • Perform a reset on the DP slave.
1	**1:** DP slave is not ready for data exchange.	• Wait for the slave to complete start-up.
2	**1:** Configuration data sent by DP master to the DP slave is inconsistent with slave configuration.	• Was the software set for the correct station type or DP slave configuration?
3	**1:** Diagnostic interrupt, generated by a RUN-STOP transition on the CPU or by the SFB 75 **0:** Diagnostic interrupt, generated by a STOP-RUN transition on the CPU or by the SFB 75	• You can read the diagnostic data.
4	**1:** Function is not supported; for example, changing the DP address using software	• Check configuration data.
5	**0:** The bit is always "0".	• -
6	**1:** DP slave type inconsistent with software configuration.	• Was the software set for the right station type? (parameter assignment error)
7	**1:** DP slave was assigned parameters by a DP master other than the DP master currently accessing the DP slave.	• The bit is always 1 if, for example, you are currently accessing the DP slave via the programming device or a different DP master. The DP address of the parameter assignment master is in the "master PROFIBUS address" diagnostic byte.

Station Status 2

Table 10- 18 Structure of station status 2 (Byte 1)

Bit	Meaning
0	**1:** The DP slave requires new parameters and configuration.
1	**1:** A diagnostic message was received. The DP slave cannot resume operation until the error has been cleared (static diagnostic message).
2	**1:** This bit is always "1" if a DP slave exists with this DP address.
3	**1:** The watchdog monitor is enabled on this DP slave.
4	**1:** DP slave has received control command "FREEZE".
5	**1:** DP slave has received control command "SYNC".
6	**0:**The bit is always set to "0".
7	**1:** DP slave is disabled, that is, it has been excluded from cyclic processing.

Station Status 3

Table 10- 19 Structure of station status 3 (Byte 2)

Bit	Meaning
0 to 6	**0:** These bits are always "0"
7	**1:** There are more diagnostic messages than the DP slave can save. The DP master cannot enter all diagnostic messages sent from the DP slave in its diagnostic buffer.

Master PROFIBUS address

The "Master PROFIBUS address" diagnostic byte stores the DP address of the DP master:

- that has configured the DP slave and
- that has read and write access to the DP slave

Table 10- 20 Structure of the Master PROFIBUS address (byte 3)

Bit	Meaning
0 to 7	DP address of the DP master that has assigned parameters to the DP slave and has read/write access to that DP slave.
	FF_H: DP slave was not configured by a DP master

Device ID

The device ID is a manufacturer ID containing a code which specifies the type of the DP slave.

Table 10- 21 Structure of the device ID (bytes 4, 5)

Byte 4	Byte 5	Device ID
81_H	99_H	IM151-7 CPU

Structure of the configured address range diagnostics of the IM151-7 CPU

Module diagnostics indicate the configured address range of intermediate memory that has received an entry.

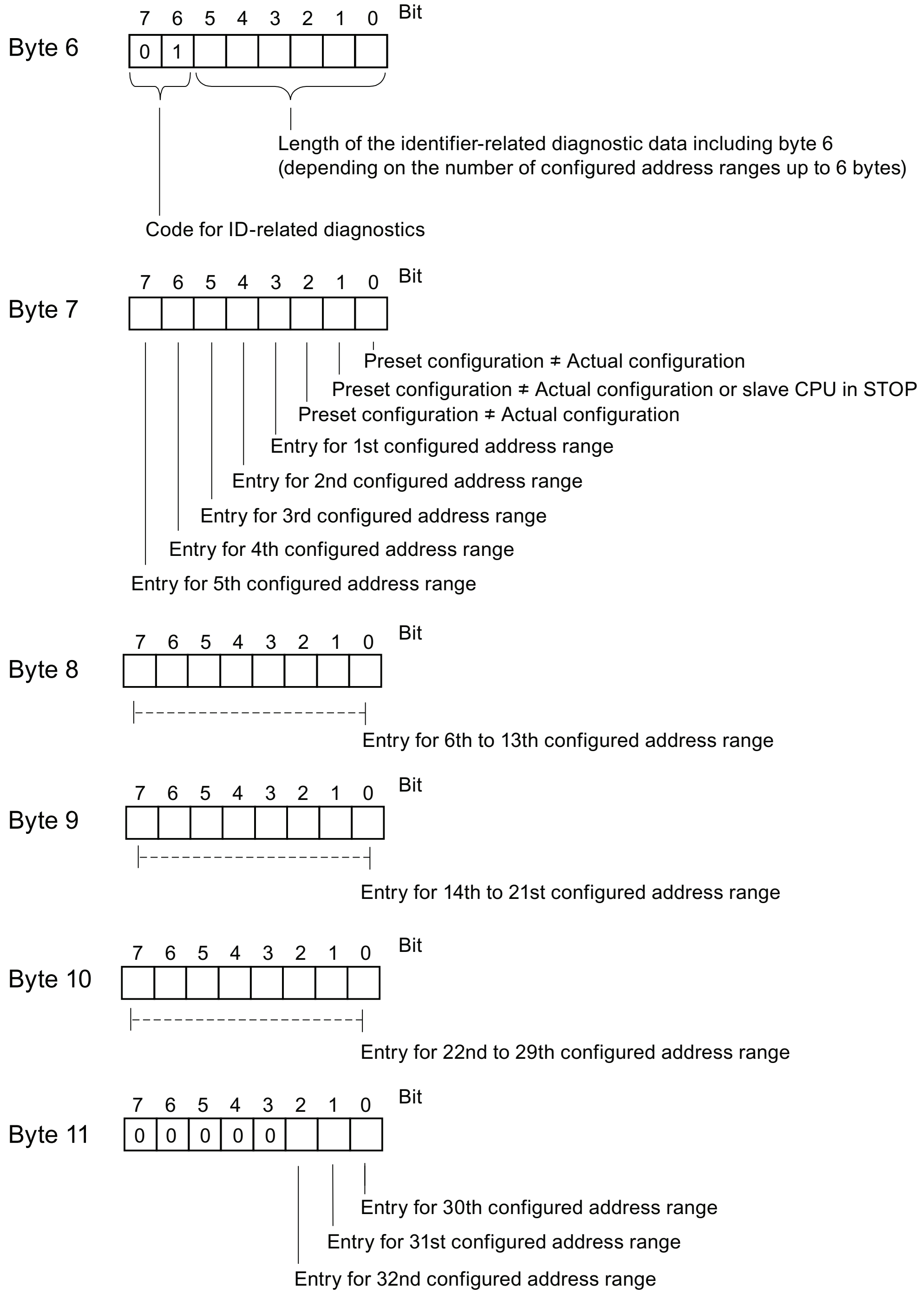


Figure 10-5 Identifier-related diagnostic data

Structure of the module status

The module status reflects the status of the configured address ranges, and provides detailed ID-specific diagnostics with respect to the configuration. The module status follows the identifier-related diagnostics and consists of a maximum of 13 bytes.

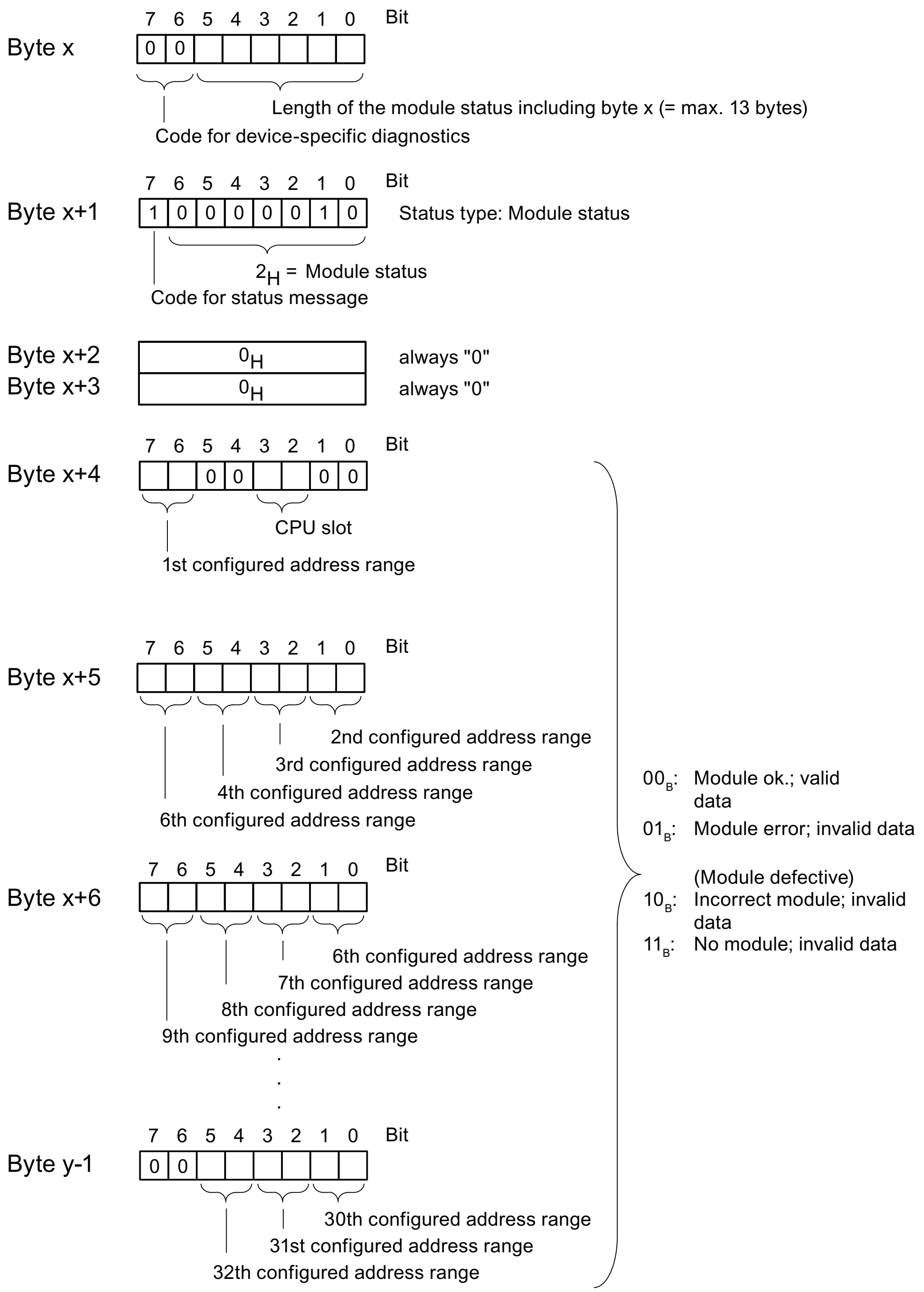


Figure 10-6 Structure of the module status

Structure of the interrupt status

The interrupt status of module diagnostics provides details on a DP slave. Device-related diagnostics starts at byte y and has a maximum length of 20 bytes.

The following figure describes the structure and content of the bytes for a configured address range of transfer memory.

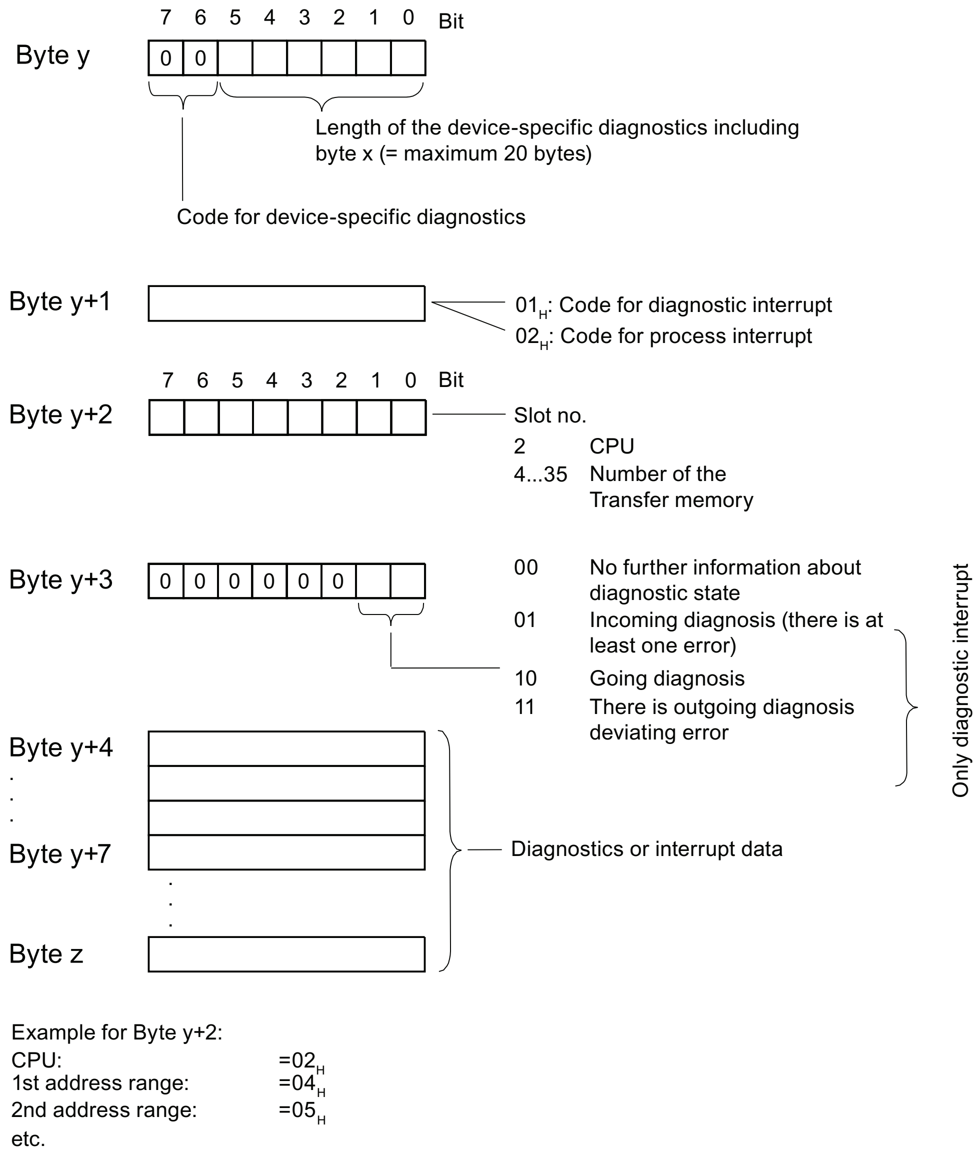


Figure 10-7 Device-specific diagnostics

Structure of the interrupt data for a process interrupt (from byte y+4)

During a hardware interrupt (in byte y+1, code 02_H stands for hardware interrupt), as of byte y+4, the four bytes of interrupt information which you transferred in the I-slave using SFC 7 "DP_PRAL" or SFB 75 "SALRM" during generation of the hardware interrupt for the master are transferred.

Structure of the interrupt data when a diagnostic interrupt is generated in response to an operating status change by the intelligent slave (after byte y+4)

Byte y+1 contains the code for a diagnostic interrupt (01_H). The diagnostic data contains the 16 bytes of status information from the CPU. The figure below shows the allocation of the first four bytes of diagnostic data. The next 12 bytes are always 0.

The data in these bytes correspond to the contents of data record 0 of diagnostic data in *STEP 7* (in this case, not all bits are used).

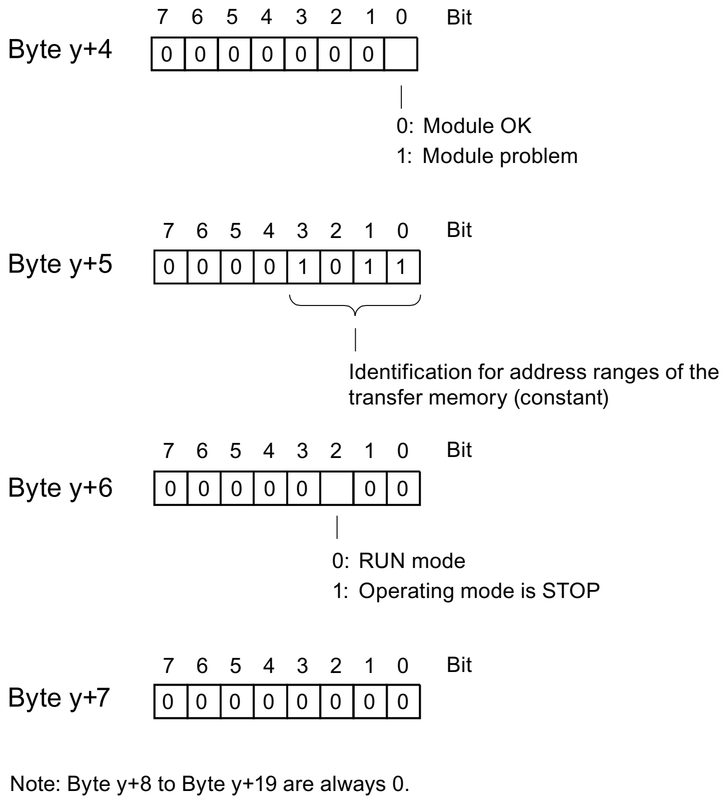


Figure 10-8 Bytes y+4 to y+7 for a diagnostic interrupt (operating status change by intelligent slave)

Structure of the interrupt data when a diagnostic interrupt is generated by SFB 75 in the I slave (as of byte y+4)

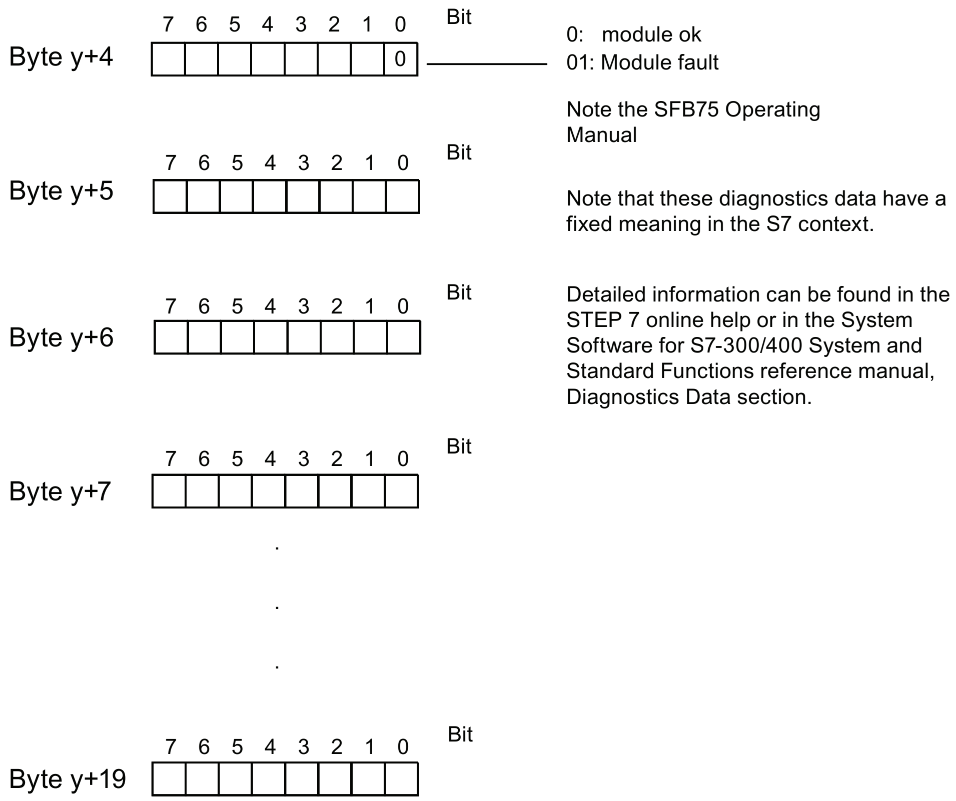


Figure 10-9 Bytes y+4 to y+19 for the diagnostics interrupt (SFB75)

10.7 Defective configuration statuses of the ET 200S

Fault indications in the diagnostics buffer

Defective configuration statuses of the ET 200S distributed I/O system are entered in the diagnostics buffer.

Error type	Error location	Cause of error	Remedy
1	04 to 66 (slot) If applicable 67 (bus terminator module)	**Communication interruption** Displays the first slot at which no I/O module is recognized. • Missing I/O module during POWER ON or several I/O modules are missing during operation. • Interruptions at the rear panel bus • Short-circuit at the rear panel bus ("04" is output as the slot). • Termination module missing. If the termination module is missing, the number of the inserted I/O modules + 1 are output.	Check the configuration of the ET 200S.

10.8 Failure of the load voltage from the power module

Load voltage failure

Should the load voltage of the power module fail, the electronic modules will behave as follows:

- No output for output modules.
- Substitute values are generated for input modules.

Note

Electronic modules that are re-parameterized during operation must be parameterized yet again once the load voltage has been restored to the power module.

---

11 Technical data

11.1 General technical data

Reference

The IM151-7 CPU interface module and the DP master module conform to the standards and test values that apply to the ET 200S distributed I/O device. Detailed information on the general technical specifications can be found in the ET 200S Distributed I/O Device (http://support.automation.siemens.com/WW/view/en/1144348) Operating Instructions.

11.2 IM151-7 CPU Interface Module

11.2.1 Block diagram IM 151-7 CPU with DP master module

The following figure shows the block diagram of the IM151-7 CPU interface module with the optional DP master module.

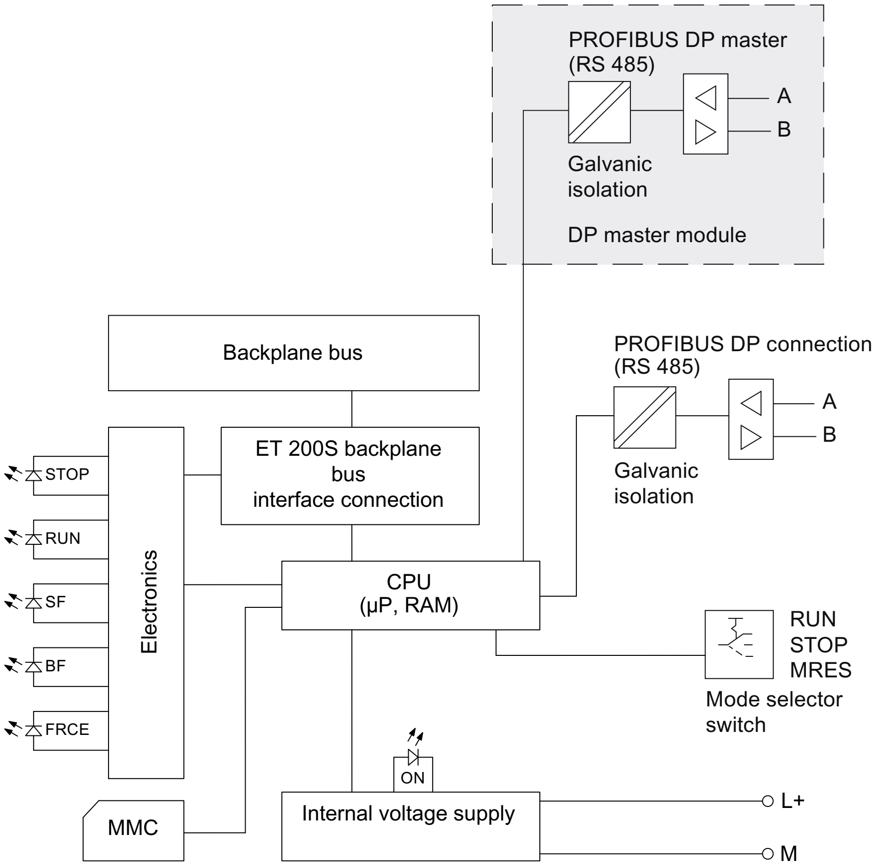


Figure 11-1 Block diagram of the IM151-7 CPU interface module with DP master module

11.2.2 Technical specifications IM151-7 CPU

Table 11- 1 Technical specifications of the IM151-7 CPU interface module

Technical specifications	
IM151-7 CPU and version	
Order number	6ES7151-7AA21-0AB0
• Hardware version	01
• Firmware version	V 3.3
• Associated programming package	*STEP 7* as of V5.2 + SP1 + HSP219 or V5.5 + SP1
Dimensions and weight	
Dimensions W x H x D (mm)	60 x 119.5 x 75
Weight	ca. 200 g
Memory	
Work memory	
• Work memory	128 KB
• Expandable	No
• Capacity of the retentive memory for retentive data blocks	64 KB
Load memory	
• Pluggable (MMC)	Yes
• Pluggable (MMC), max.	8 MB
Buffering	Yes (ensured with SIMATIC Micro Memory Card - maintenance-free)
Data retention on the SIMATIC Micro Memory Card (after final programming)	At least 10 years
Execution times	
Processing times of	
• Bit operations	0.06 µs
• Word instructions	0.12 µs
• Fixed-point arithmetic	0.16 µs
• Floating-point arithmetic	0.59 µs
Timers / counters and their retentivity	
S7 counters	
• Number	256
• Retentivity, configurable	Yes
• Retentivity, preset	From C0 to C7
• Counting range	0 to 999

Technical specifications	
IEC Counters	
• Available	Yes
• Type	SFB
• Number	Unlimited (limited only by work memory)
S7 timers	
• Number	256
• Retentivity, configurable	Yes
• Retentivity, preset	Not retentive
• Timer range	10 ms to 9990 s
IEC timers	Yes
• Type	SFB
• Number	Unlimited (limited only by work memory)
Data areas and their retentive address areas	
Bit memory	
• Number, max.	256 bytes
• Retentivity, configurable	Yes, from MB0 to MB255
• Retentivity, preset	From MB0 to MB15
Number of clock memories	8 (1 memory byte)
Local data per priority class, max.	32 KB per runtime level / 2 KB per block
Blocks	
Total number of blocks	1024 (DBs, FCs, FBs) The maximum number of blocks that can be loaded may be reduced if you are using another SIMATIC Micro Memory Card.
OBs	See S7300-CPUs and ET 200-CPUs Instruction List (http://support.automation.siemens.com/WW/view/en/31977679)
• Size, max.	64 KB
• Number of free-cycle OBs	1 (OB 1)
• Number of time-of-day interrupt OBs	1 (OB 10)
• Number of time-delay interrupt OBs	2 (OB 20, 21)
• Number of cyclic interrupt OBs	4 (OB 32, OB 33, OB 34, OB 35)
• Number of hardware interrupt OBs	1 (OB 40)
• Number of DPV1 interrupt OBs	3 (OB 55, 56, 57)
• Number of asynchronous error OBs	6 (OB 80, 82, 83, 85, 86, 87) (OB 83 only for centralized I/O, not for DP)
• Number of startup OBs	1 (OB 100)
• Number of synchronous error interrupt OBs	2 (OB 121, 122)

Technical specifications	
Nesting depth	
• Per priority class	16
• Additional within an error OB	4
FBs	See *Instruction List*
• Number, max.	1024 (in the number range 0 to 7999)
• Size	64 KB
FCs	See *Instruction List*
• Number, max.	1024 (in the number range 0 to 7999)
• Size	64 KB
Data blocks	
• Number, max.	1024 (in the number range 1 to 16000)
• Size, max.	64 KB
• Non-retain support (configurable retentive address areas)	Yes
Address areas (I/O)	
Total I/O address area	
• Inputs, freely adressable	2048 bytes
• Outputs, freely adressable	2048 bytes
• Of which distributed – Inputs, freely adressable – Outputs, freely adressable	 2048 bytes 2048 bytes
Process I/O image	
• Inputs, adjustable	2048 bytes
• Outputs, adjustable	2048 bytes
• Inputs, preset	128 bytes
• Outputs, preset	128 bytes
Digital channels	
• Inputs	16336
• Outputs	16336
• Inputs, central	496
• Outputs, central	496
Analog channels	
• Inputs	1021
• Outputs	1021
• Inputs, central	124
• Outputs, central	124

Technical specifications	
Removal	
Mounting rail	1
I/O module for each ET 200S	Max. 63
Station width	≤ 1 m or < 2 m
Current carrying capacity per load group (power module)	Max. 10 A
Time-of-day	
Clock	
• Hardware clock (real-time clock)	Yes
• Factory setting	DT#1994-01-01-00:00:00
• Buffered, can be synchronized	Yes
• Buffered period	Typically 6 weeks (at an ambient temperature of 40 °C)
• Behavior of the clock on expiration of the buffered period	The clock keeps running, continuing at the time-of-day it had when power was switched off.
• Behavior of the real-time clock after POWER ON	The clock continues running after POWER OFF.
• Deviation per day	typ. 2 s, max. 10 s
Operating hours counter	
• Number	1
• Number	0
• Value range	0 to 2^{31} hours (using the SFC 101)
• Granularity	1 hour
• Retentive	Yes; must be manually restarted after every restart
Time synchronization	
• Supported	Yes
• On MPI, time master	Yes
• On MPI, time slave	Yes
• On DP, time master	Yes (DP slave must be time slave)
• On DP, time slave	Yes
• On the AS, time master	No
• On the AS, time slave	No

Technical specifications	
S7 message functions	
Number of stations that can be logged on for signaling functions	12 (depends on the number of connections configured for programming device / OP communication)
Process diagnostics messages	
• Supported	Yes
• Simultaneously enabled interrupt S blocks, max.	300
Test and startup functions	
Status/control	
• Monitor/modify variable	Yes
• Variables	Inputs, outputs, memory bits, DBs, timers, counters
• Maximum number of variables	30
• Number of variables, of those status variables, max.	30
• Number of variables, of those modify variables, max.	14
Force	
• Force	Yes
• Forcing, variables	Inputs/Outputs
• Maximum number of variables	10
Block status	Yes; (max. 2 blocks simultaneously)
Single-step	Yes
Number of breakpoints	4
Diagnostics buffer	
• Yes	Yes
• Maximum number of entries	500
• Adjustable	No
• of which are power-failure-proof	The last 100 entries are retained.
• Maximum number of entries that can be read in RUN – Adjustable – Default	499 Yes (from 10 to 499) 10
Service data can be read	Yes
Monitoring functions	
Status LEDs	Yes

Technical specifications	
Communication functions	
PG/OP communication	Yes
Prioritized OCM communication	
• Supported	No
Routing	Yes (with DP master module)
• Connections, max.	4
Data set routing	Yes (with DP master module)
Global data communication	
• Supported	Yes
• Number of GD circles, max.	8
• Number of GD packages, max. – Senders, max. – Receivers, max.	8 8 8
• Size of GD packages, max. – of which consistent, max.	22 bytes 22 bytes
S7 basic communication	
• Supported	Yes
• User data per job, max.	76 bytes
• User data per job (consistent), max.	76 bytes (for X-SEND/REC); 64 bytes (for X-PUT/GET as server)
S7 communication	
• As server	Yes
• As client	No
• User data per job, max. • User data per job (consistent), max.	See *STEP 7 online help*, common parameters of SFBs/FBs and SFC/FC for S7 communication
Number of connections	
• Total	12
Suitable for PG communication	11
• PG communication, reserved	1
• PG communication, configurable, min.	1
• PG communication, configurable, max.	11
Suitable for OP communication	11
• OP communication, reserved	1
• OP communication, configurable, min.	1
• OP communication, configurable, max.	11

Technical specifications	
Suitable for S7 basic communication	10
• S7 basic communication, reserved (default)	0
• S7 basic communication, configurable, min.	0
• S7 basic communication, configurable, max.	10
1st interface	
Port designation	X1
Type of interface	Integrated RS 485 interface
Physics	RS 485
Isolated	Yes
Max. interface power supply (15V DC to 30V DC)	80 mA
Connection	9-pin sub-D socket
Functionality	
MPI	Yes
DP master	No
DP slave	Yes (active/passive)
Point-to-point link	No
MPI	
Services	
PG/OP communication	Yes
Routing	Yes (with DP master module)
Global data communication	Yes
S7 basic communication	Yes
S7 communication, as server	Yes
S7 communication, as client	No
Transmission rate, max.	12 Mbps
DP slave	
Services	
PG/OP communication	Yes
Routing	Yes (only with active, integrated DP slave interface and installed DP master module in DP master operation)
Global data communication	No
S7 basic communication	No
S7 communication	Yes (only server; connection configured at one end)
Direct data exchange	Yes
DPV1	No
Transmission rate, max.	12 Mbps
Automatic baud rate detection	Yes (only if interface is passive)

Technical specifications	
Transfer memory	
• Inputs	244 bytes
• Outputs	244 bytes
Address range, max.	32
• User data per address range, max.	32 bytes
GSD file	The current GSD file is available for download from the Internet (http://support.automation.siemens.com/WW/view/en/10805317/133100).
2nd interface	
Port designation	X2; on the DP master module
Type of interface	RS 485, integrated interface on the DP master module
Physics	RS 485
Isolated	Yes
Max. interface power supply (15V DC to 30 V)	No
Connection	9-pin sub-D socket
Functionality	
MPI	No
DP master	Yes
DP slave	No
DP master	
Services	
PG/OP communication	Yes
Routing	Yes
Data set routing	Yes (to field devices on the PROFIBUS DP)
Global data communication	No
S7 basic communication	Yes (intelligent blocks only)
S7 communication	Yes (only server; connection configured at one end)
Transmission rate, max.	Up to 12 Mbps
Support for constant bus cycle time	Yes
Isochronous mode	No
Enable/disable DP slaves	Yes
• Max. number of DP slaves that can be enabled / disabled simultaneously	8
SYNC/FREEZE	Yes
Direct data exchange	Yes
DPV1	Yes
Number of DP slaves, max.	32

Technical specifications	
Address area	
• Inputs, max.	2048 bytes
• Outputs, max.	2048 bytes
User data per DP slave	
• Inputs, max.	244 bytes
• Outputs, max.	244 bytes
Programming	
• LAD	Yes
• FBD	Yes
• STL	Yes
• SCL	Yes
• CFC	Yes
• GRAPH	Yes
• HiGraph	Yes
Instruction set	See *Instruction List*
Nesting levels	8
System functions (SFC)	See *Instruction List*
System function blocks (SFB)	See *Instruction List*
User program protection / password protection	Yes
Encryption of blocks	Yes; with S7 Block Privacy
Voltages, currents, electrical potentials	
Rated supply voltage for the electronic components 1L+	24 VDC
• Permissible range	19.2 V to 28.8 V
• Reverse polarity protection	Yes; against destruction
• Short-circuit protection	Yes
• Power failure buffering	5 ms
Current consumption from rated supply voltage 1L+	
• IM151-7 CPU	Typically 320 mA
• IM151-7 CPU + DP master module	Typically 410 mA
• Power supply for the ET 200S backplane bus	Max. 700 mA
Inrush current	Typically 1.8 A
I^2t	Typically 0.09 A^2s

Technical specifications	
External fusing of power supply lines (recommended)	
• Electronic / encoder supply 1L+	24 VDC / 16 A circuit-breaker with type B or C tripping characteristic Note: A 24V DC/16A circuit-breaker with type B tripping characteristic trips before the equipment fuse is tripped. A 24 VDC/16A circuit-breaker with type C tripping characteristic trips after the equipment fuse is tripped.
Power loss	Typically 4.2 W
Insulation tested with	500 VDC
Galvanic isolation	
• between the backplane bus and supply voltages 1L+	No
• between the electronics and supply voltage 1L+	No
Maximum potential difference	75 VDC, 60 VAC
Status, interrupts, diagnostics	
Interrupts	Yes
Diagnostics function	Yes
Group errors	Red "SF" LED
Monitoring of the supply voltage for the electronic components 1L+	Green LED "ON"

11.3 DP master module

11.3.1 Technical specifications - DP master module

Table 11- 2 Technical specifications of the DP master module

Technical specifications	
Removal	
DP master module	
• Position	To the right of IM151-7 CPU
• Number per IM151-7 CPU	1
Dimensions	
Mounting dimensions W x H x D (mm)	35 x 119.5 x 75
Weight	ca. 100 g
Status, interrupts, diagnostics	
PROFIBUS DP bus monitoring	Red "BF" LED

Appendix A

A.1 Order numbers

A.1.1 Module order numbers

IM151-7 CPU interface module

Table A- 1 IM151-7 CPU order numbers

Designation	Order number
IM151-7 CPU interface module with terminating module, 1 unit *	6ES7151-7AA21-0AB0
* The SIMATIC Micro Memory Card is not supplied as standard.	

DP master module

Table A- 2 DP master module order numbers

Designation	Order number
DP master module, 1 unit	6ES7138-4HA00-0AB0

A.1.2 Order numbers of accessories

IM151-7 CPU interface module accessories

Table A- 3 IM151-7 CPU accessories order numbers

Designation	Order number
SIMATIC Micro Memory Card 64k	6ES7953-8LFxx-0AA0
SIMATIC Micro Memory Card 128k	6ES7953-8LGxx-0AA0
SIMATIC Micro Memory Card 512k	6ES7953-8LJxx-0AA0
SIMATIC Micro Memory Card 2M	6ES7953-8LLxx-0AA0
SIMATIC Micro Memory Card 4M	6ES7953-8LMxx-0AA0
SIMATIC Micro Memory Card 8M (suitable for an FW update)	6ES7953-8LPxx-0AA0
Label sheets DIN A4, 10 units	
• beige	6ES7193-4BA00-0AA0
• Yellow	6ES7193-4BB00-0AA0
• red	6ES7193-4BD00-0AA0
• petrol	6ES7193-4BH00-0AA0

Connectors and cables

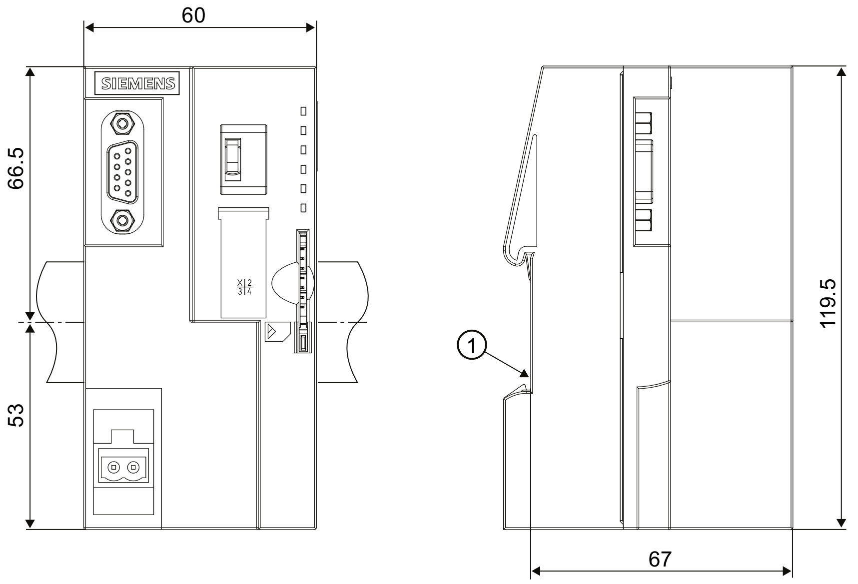
Table A- 4 Connectors and cables order numbers

Designation	Order number
PROFIBUS	
PROFIBUS bus connector (12 Mbps), 90° cable outlet	
• without PG connection socket	6ES7972-0BA12-0XA0
• with PG connection socket	6ES7972-0BB12-0XA0
PROFIBUS bus connector (12 Mbps) with FastConnect connection system, 90° cable outlet	
• without PG connection socket	6ES7972-0BA52-0XA0
• with PG connection socket	6ES7972-0BB52-0XA0
PROFIBUS bus connector (12 Mbps), 35° cable outlet	
• without PG connection socket	6ES7972-0BA42-0XA0
• with PG connection socket	6ES7972-0BB42-0XA0
PROFIBUS bus connector (12 Mbps) with FastConnect connection system, 35° cable outlet	
• without PG connection socket	6ES7972-0BA60-0XA0
• with PG connection socket	6ES7972-0BB60-0XA0
PROFIBUS FC cable Sold by meter, min. ordering quantity 20 m Delivery unit max. 1000 m, 1 m	
• FC Standard Cable	6XV1830-0EH10
• FC Trailing Cable (for cable carriers)	6XV1830-3EH10
• FC Food Cable (PE sheath)	6XV1830-0GH10
• FC Food Cable (PUR sheath)	6XV1830-0JH10
PROFIBUS FastConnect stripping tool	6GK1905-6AA00

A.2 Dimension drawings

A.2.1 IM151-7 CPU Interface Module

IM151-7 CPU interface module

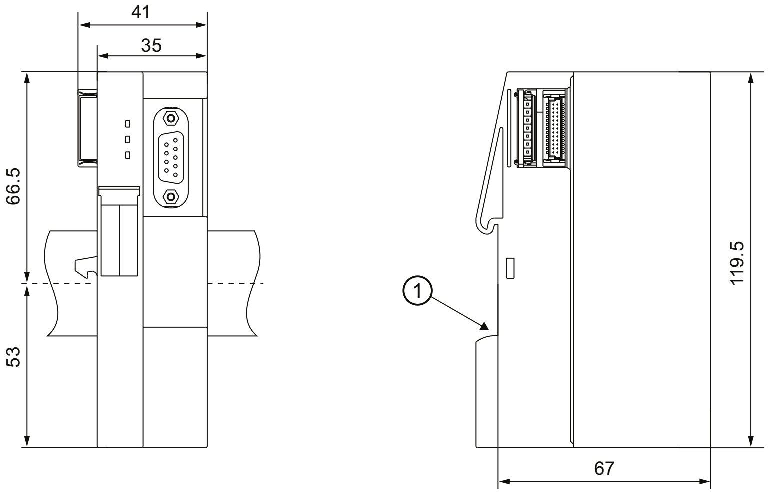


① Mounting rail rest

Figure A-1 Dimension drawing IM151-7 CPU interface module

A.2.2 DP master module

DP master module



① Mounting rail rest

Figure A-2 Dimension drawing DP master module

A.3 Cycle and response times

A.3.1 Overview

Overview

This section contains detailed information about the following topics:

- Cycle time
- Response time
- Interrupt response time

Reference: Cycle time

You can view the cycle time of your user program on the programming device. For additional information, refer to the *STEP 7 Online Help* or to the Configuring Hardware and Connections in STEP 7 (http://support.automation.siemens.com/WW/view/en/45531110) manual.

Reference: Execution time

Execution times are listed in the S7300-CPUs and ET 200-CPUs Instruction List (http://support.automation.siemens.com/WW/view/en/31977679). The instruction list contains the execution times in table form for all

- *STEP 7* instructions processed by the IM151-7 CPU interface module,
- SFCs / SFBs integrated in the IM151-7 CPU interface module,
- The IEC functions that can be called in *STEP 7*.

A.3.2 Cycle time

A.3.2.1 Overview: Cycle time

Introduction

This section explains what we mean by the term "cycle time", what it consists of, and how you can calculate it.

Meaning of the term cycle time

The cycle time represents the time that an operating system needs for one program pass, i.e. one OB 1 cycle, including all program sections and system activities interrupting this cycle. This time is monitored.

Time slice model

Cyclic program processing, and therefore user program execution, is based on time shares. To clarify these processes, let us assume that every time share has a length of precisely 1 ms.

Process image

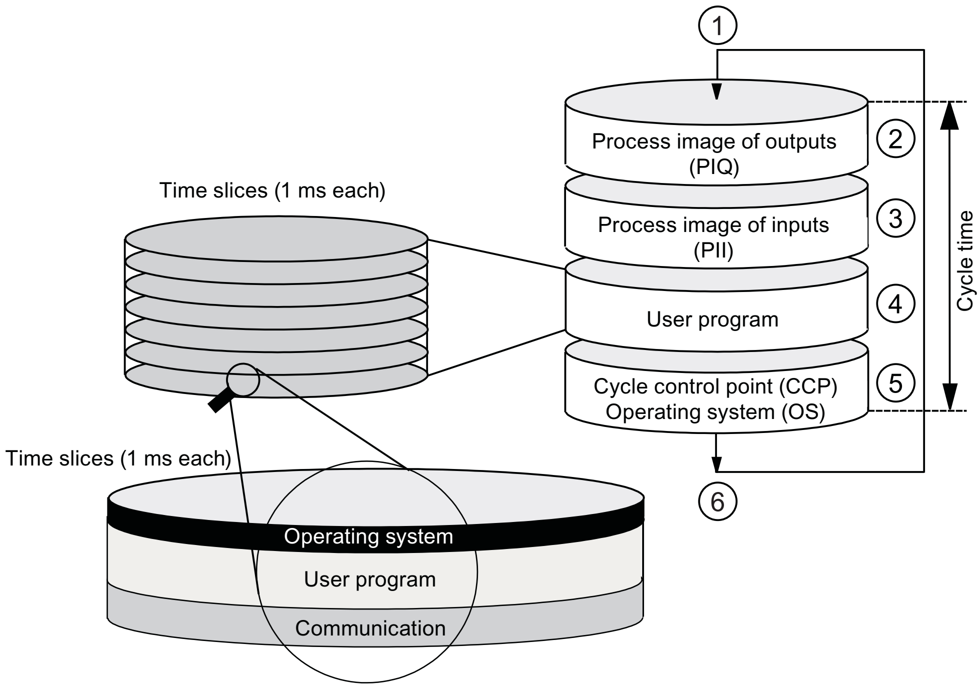
During cyclic program processing, the IM151-7 CPU interface module requires a consistent image of the process signal. To ensure this, the process signals are read / written prior to program execution. The IM151-7 CPU interface module then does not address input (I) and output (Q) operand areas directly at the I/O modules, but rather accesses the system memory area of the IM151-7 CPU interface module containing the I/O process image.

Sequence of cyclic program processing

The table and figure below show the phases in cyclic program processing.

Table A- 5 Cyclic program processing

Step	Sequence
1	The operating system initiates cycle time monitoring.
2	The IM151-7 CPU interface module writes the values from the process output image to the output modules.
3	The IM151-7 CPU interface module reads the status at the inputs of the input modules and then updates the process input image.
4	The IM151-7 CPU interface module processes the user program in time slices and executes the operations specified in the program.
5	At the end of a cycle, the operating system executes queued tasks, for example, loading and deleting blocks.
6	The IM151-7 CPU interface module then returns to the start of the cycle, and restarts the cycle time monitoring.



PIQ: Process image of outputs
PII: Process image of inputs
CCP: Cycle check point
OS: Operating system

In contrast to S7-400 CPUs, the IM151-7 CPU interface module only accesses data with an OP / TP (monitor and modify functions) at the cycle control point (for data consistency, see section Technical data (Page 179)). Processing of the user program is not interrupted by the monitor and modify functions.

Extending the cycle time

Always make allowances for the extension of the cycle time of a user program due to:

- Time-based interrupt processing
- Process interrupt processing
- Diagnostics and error processing
- Processing isochronic interrupts
- Communication with programming devices (PGs) and Operator Panels (OPs)
- Testing and commissioning such as, e.g. status/controlling of variables or block status functions
- Transfer and deletion of blocks, compressing user program memory
- Write/read access to the SIMATIC Micro Memory Card using SFC 82 to 84 in the user program

A.3.2.2 Calculating the cycle time

Introduction

The cycle time is derived from the sum of the following influencing factors.

Process image update

The table below shows the times that an IM151-7 CPU interface module needs to update the process image (process image transfer time). The specified times may be extended as a result of interrupts or IM151-7 CPU interface module communication. The process image transfer time is calculated as follows:

Table A- 6 Formula for calculating the typical transfer time for the process image (PI)

The transfer time of the process image is calculated as follows:	
Base load K	+ Number of bytes in the PI for ET 200S I/Os (A)
	+ Number of words in the PI via PROFIBUS DP (D)
	= Transfer time for the process image

Table A- 7 Data for calculating the process image (PI) transfer time

Constant	Components	IM151-7 CPU
C	Base load	140 μs
A	per byte in the PO for centralized ET 200S I/Os	60 μs
D (PROFIBUS DP only)	per word in the DP area for the DP interface integrated in the DP master	0.5 μs

Extending the user program processing time

In addition to actually working through the user program, the operating system of your IM151-7 CPU interface module also runs a number of processes in parallel, such as timer management for the core operating system. These processes extend the processing time of the user program by up to 10%.

Operating system processing time at the cycle control point

The table below shows the operating system processing time at the cycle control point of the IM151-7 CPU interface module. This time applies without:

- Testing and commissioning routines, e.g. status/controlling of variables or block status functions
- Transfer and deletion of blocks, compressing user program memory
- Communication
- Writing, reading of the SIMATIC Micro Memory Card with SFC 82 to 84

Table A- 8 Typical operating system processing time at the cycle control point

Interface module	Cycle control at the cycle control point (CCP)
IM151-7 CPU	150 µs

Extension of the cycle time as a result of nested interrupts

Enabled interrupts also extend cycle time. Details are found in the table below.

Table A- 9 Typical extended cycle time due to nested interrupts

Interrupt type	Process interrupt	Diagnostic interrupt	Time-of-day interrupt	Delay interrupt	Watchdog interrupt
IM151-7 CPU	250 µs	250 µs	300 µs	200 µs	170 µs

The program runtime at interrupt level must be added to this time extension.

Extension of the cycle time due to error

Table A- 10 Typical cycle time extension as a result of errors

Type of error	Programming errors	I/O access errors
IM151-7 CPU	120 µs	130 µs

You have to add the program execution time of the interrupt OB to this increase. The times required for multiple nested interrupt/error OBs are added accordingly.

A.3.2.3 Communication load

Configured communication load for PG/OP communication and S7 communication

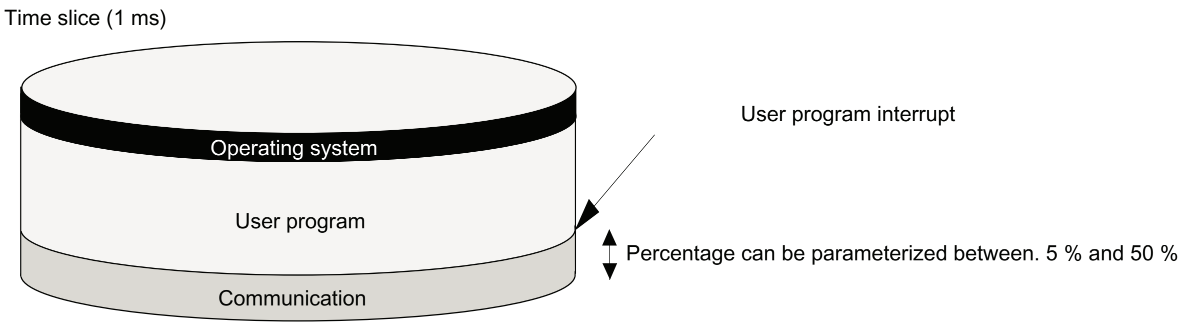
The CPU operating system continuously provides a specified percentage of total CPU processing performance (time slice technology) for communication tasks. Processing performance not required for communication is made available to other processes.

In the hardware configuration you can specify a communication load value between 5% and 50%. The default value is 20%.

The extension of the cycle time is dependent on the communication loading and can fluctuate.

You can use the following formula for calculating the maximum cycle time extension factor:

100 / (100 – configured communication load in %)



Example: 20% communication load

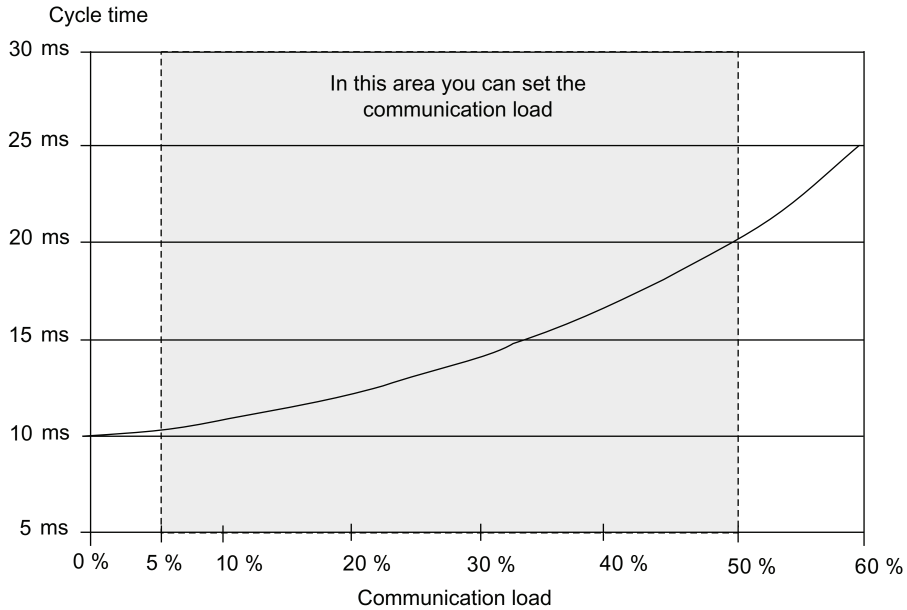
In your hardware configuration, you have specified a communication load of 20%. The calculated cycle time is 10 ms. Using the above formula, the cycle time is extended by the factor 1.25.

Example: 50% communication load

In your hardware configuration, you have specified a communication load of 50%. The calculated cycle time is 10 ms. Using the above formula, the cycle time is extended by the factor 2.

Dependency of actual cycle time on the communication load

The figure below describes the non-linear dependency of the actual cycle time on the communication load. In our example we have chosen a cycle time of 10 ms.



Influence on the actual cycle time

From the statistical viewpoint, asynchronous events such as interrupts occur more frequently within the OB1 cycle when the cycle time is extended as a result of communication load. This further extends the OB1 cycle. This extension depends on the number of events that occur per OB1 cycle and the time required to process these events.

Note

Change the value of the "communication load" parameter to check the effects on the cycle time during system runtime. You must consider the communication load when setting the maximum cycle time, otherwise time errors may occur.

Tips

- Use the default setting whenever possible.
- Increase this value only if the CPU is used primarily for communication and if the user program is not time critical.
- In all other situations you should only reduce this value.

A.3.2.4 Cycle time extension as a result of test and commissioning functions

Runtimes

The runtimes of the testing and commissioning functions are operating system runtimes, so they are the same for every CPU. How the cycle time is extended as a result of active testing and commissioning functions is shown in the table below.

Table A- 11 Cycle time extension as a result of test and commissioning functions

Function	IM151-7 CPU
Status variable	Negligible
Control variable	Negligible
Status block	Typ. 3 μs for each monitored line + 3 x runtime of monitored block The monitoring of large blocks and the monitoring of loops can lead to a significant increase in the cycle time.

Setting process and test mode in the LAD/FBD/STL editor

Switching between process and test mode is carried out directly in the LAD/FBD/STL editor in the "Test/Mode" menu.

Loops in the test and process mode are handled differently in the Status block.

- **Process mode:** First loop iteration is displayed
- **Test mode:** Last loop iteration is displayed. Leads to a significant cycle time increase for many loop iterations.

In terms of function, there is also no difference between process mode and test mode.

Note

It is also possible to set breakpoints in test mode.

A.3.3 Response time

A.3.3.1 Overview: Response time

Definition of response time

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

Fluctuation width

The physical response time lies between the shortest and the longest response time. You must always reckon with the longest response time when configuring your system.

The shortest and longest response times are shown below, to give you an idea of the fluctuation width of the response time.

Factors

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

- Delay in the I/O module inputs and outputs
- Additional DP cycle times on PROFIBUS DP
- Execution in the user program

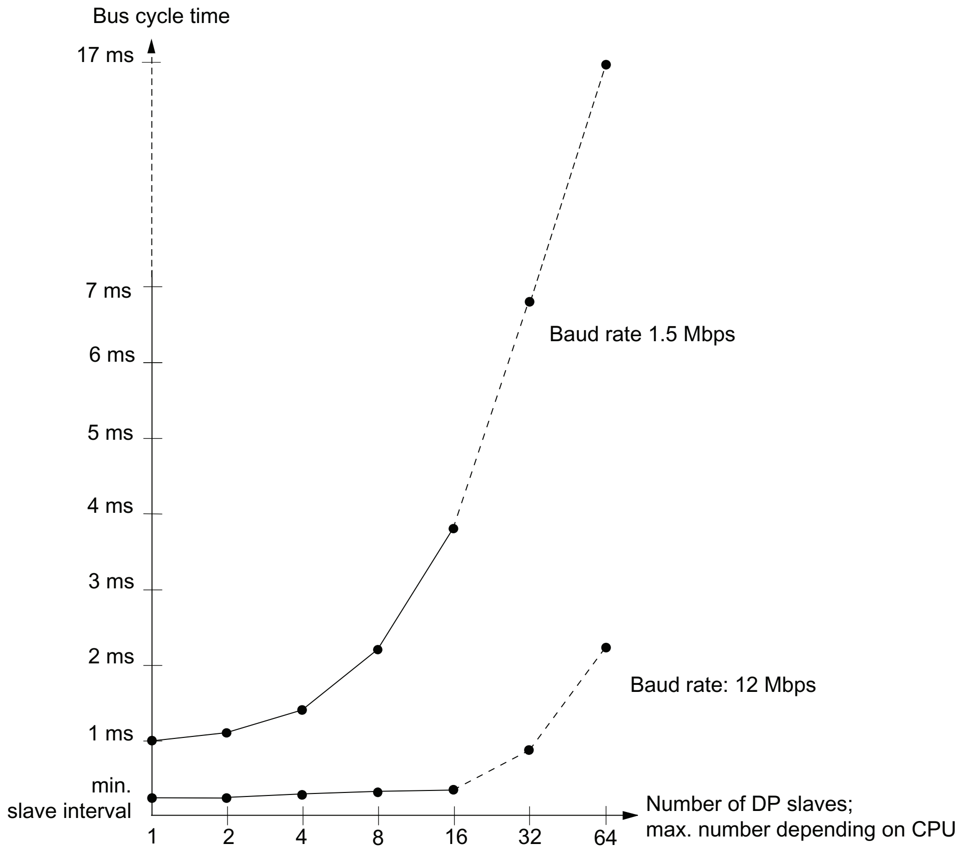
Reference

The delay times are described in the technical data for the I/O modules in the ET 200S Distributed I/O Device (http://support.automation.siemens.com/WW/view/en/10805258/133300) manual.

DP cycle times in the PROFIBUS DP network

If you configured your PROFIBUS DP master system with *STEP 7*, then *STEP 7* will calculate the typical DP cycle time that must be expected. You can then view the DP cycle time of your configuration on the programming device.

The figure below gives you an overview of the DP cycle time. In this example, let us assume that the data of each DP slave has an average length of 4 bytes.

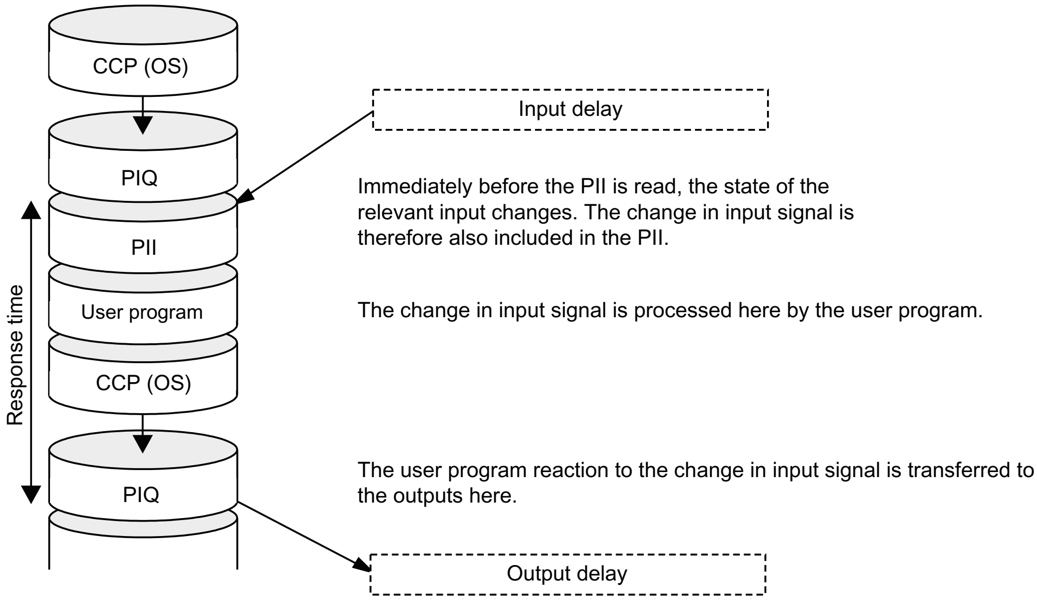


With multi-master operation on a PROFIBUS-DP network, you must make allowances for the DP cycle time at each master. That is, you will have to calculate the times for each master separately and then add up the results.

A.3.3.2 Shortest response time

Conditions for the shortest response time

The figure below shows the conditions under which the shortest response time is reached.



Calculation

The (shortest) response time is the sum of:

Table A- 12 Formula: Shortest response time

	1 x process image transfer time for the inputs
+	1 x process image transfer time for the outputs
+	1 x program processing time
+	1 × operating system processing time at the CCP
+	Delay in the inputs and outputs
=	**Shortest response time**

The result is equivalent to the sum of the cycle time plus the I/O delay times.

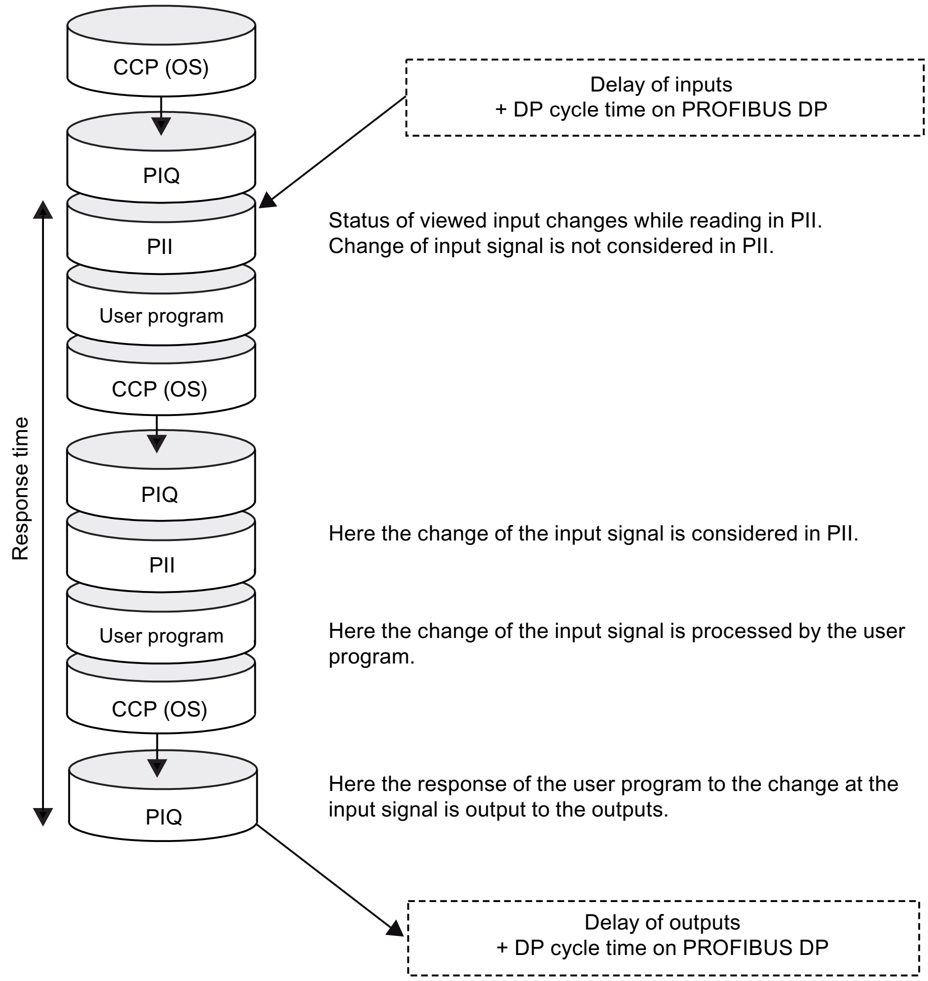
See also

Overview: Response time (Page 205)

A.3.3.3 Longest response time

Conditions for the longest response time

The figure below shows the conditions under which the longest response time is reached.



Calculation

The (longest) response time is the sum of:

Table A- 13 Formula: Longest response time

	2 x process image transfer time for the inputs
+	2 x process image transfer time for the outputs
+	2 x program processing time
+	2 × operating system processing time
+	2 x DP cycle time on PROFIBUS DP (only if PROFIBUS DP is used)
+	Delay in the inputs and outputs
=	**Longest response time**

This is equivalent to the sum of twice the cycle time and the delay in the inputs and outputs plus twice the DP cycle time on the PROFIBUS DP.

See also

Overview: Response time (Page 205)

A.3.4 Interrupt response time

A.3.4.1 Overview: Interrupt response time

Definition of interrupt response time

The interrupt response time is the time that elapses between the first occurrence of an interrupt signal and the call of the first interrupt OB instruction. Generally valid: Higher-priority interrupts take priority. This means that the interrupt response time is increased by the program processing time of the higher-priority interrupt OBs and the interrupt OBs of equal priority which have not yet been executed (queued).

Process interrupt and diagnostic interrupt response times of the IM151-7 CPU interface module

Table A- 14 Process interrupt and diagnostic interrupt response times of the IM151-7 CPU interface module

Interrupt response times (without communication) for ...	Duration
Process alarm / diagnostic alarm	less than 10 ms

Process interrupt processing

Process interrupt processing begins after process interrupt OB 40 is called. Higher-priority interrupts stop process interrupt processing. Direct I/O access is executed during runtime of the instruction. After process interrupt processing has terminated, cyclic program execution continues or further interrupt OBs of equal or lower priority are called and processed.

See also

Overview (Page 197)

A.3.4.2 Reproducibility of time-delay and watchdog interrupts

Definition of "reproducibility"

Time-delay interrupt:

The period that expires between the call of the first operation in the interrupt OB and the programmed time of interrupt.

Watchdog interrupt:

The fluctuation range of the interval between two successive calls, measured between the respective initial operations of the interrupt OB.

Reproducibility

The following times apply to the IM151-7 CPU interface module:

- Time-delay interrupt: +/- 100 µs
- Watchdog interrupt: +/- 100 µs

These times only apply if the interrupt can actually be executed at this time and if it is not delayed, for example, by higher-priority interrupts or queued interrupts of equal priority.

A.4 Additional documentation

The following documentation contains detailed information on specific topics. The appropriate areas of these operating instructions refer to this documentation.

You can find the documents on the Internet.

Name of manual	Description
Operating Instructions ET 200S Distributed I/O System (http://support.automation.siemens.com/WW/view/en/1144348)	Application planning; assembling; wiring and assembly; commissioning; functions; alarm, error, and system messages; interface modules, COMPACT modules
Manuals (http://support.automation.siemens.com/WW/view/en/10805258/133300) on the ET 200S Distributed I/O System	Description of functions and technical specifications of the terminal modules, power modules, and digital and analog electronic modules
List manual S7 300 CPU and ET 200 CPU instruction list (http://support.automation.siemens.com/WW/view/en/31977679)	• List of the instruction set of the CPUs and their execution times. • List of the executable blocks (OBs/SFCs/SFBs) and their execution times.
Reference Manual System software for S7-300/400 system and standard functions, Volume 1/2 (http://support.automation.siemens.com/WW/view/en/1214574)	Overview of objects included in the operating systems for S7-300 and S7-400 CPUs: OBs SFCs SFBs IEC functions diagnostics data system status list (SSL) events This manual is part of the *STEP 7* reference information. You can also find the description in the *Online Help for STEP 7*.
Manual Programming with STEP 7 (http://support.automation.siemens.com/WW/view/en/18652056)	This manual provides an overview of programming with *STEP 7*. This manual is part of the *STEP 7* basic information. You can also find the description in the *Online Help for STEP 7*.
Manual Configuring Hardware and Communication Connections with STEP 7 (http://support.automation.siemens.com/WW/view/en/45531110)	basics, configuration, saving, importing, exporting, networking, configuring connections, downloading
Manual CPU 31xC and CPU 31x, Technical specifications (http://support.automation.siemens.com/WW/view/en/12996906)	Description of: • Operating and display elements • Communication • Memory concept • Cycle and response times • Technical data

Name of manual	Description
Manual Communication with SIMATIC (http://support.automation.siemens.com/WW/view/en/1254686)	Description of: Basics, services, networks, communication functions, connecting PGs/OPs, project design and configuration in *STEP 7*
Manual The Process Device Manager (http://support.automation.siemens.com/WW/view/en/21407212)	Starting *SIMATIC PDM*, configuring networks and devices, working with *SIMATIC PDM*, communication, diagnostics
Manual Diagnostic repeater for PROFIBUS DP (http://support.automation.siemens.com/WW/view/en/7915183)	product overview, functions, configuration possibilities, installation, wiring, commissioning, diagnostics

Glossary

Accumulator

Accumulators represent CPU register and are used as buffer memory for download, transfer, comparison, calculation and conversion operations.

See also CPU

Address

An address is the identifier of a specific operand or operand area.
Examples: Input I 12.1; Memory Word MW 25; Data Block DB 3.

Analog modules

Analog modules convert analog process values (for example, temperature) into digital values that can be processed by the IM 151-7 CPU interface module or convert digital values into analog manipulated variables.

Application

→ *User program*

Application

An application is a program that runs directly on the MS-DOS / Windows operating system. Applications on the programming device include, for example, the *STEP 7* basic package, S7-GRAPH and others.

Automation system

An automation system is a programmable logic controller in the context of SIMATIC S7.

See also Programmable Logic Controller

Backplane bus

Serial data bus used by the interface module to communicate with electronic modules and to supply power to these. The individual modules are interconnected by means of terminal modules.

Backup memory

Backup memory ensures buffering of the memory areas of a CPU without backup battery. It backs up a configurable number of timers, counters, bit memory, data bytes and retentive timers, counters, bit memory and data bytes).

See also CPU

Baud rate

Data transfer rate (in bps)

Bit memory

Bit memory are part of the CPU's system memory. They store intermediate results of calculations. They can be accessed in bit, word or dword operations.

See System memory

Bus

A bus is a communication medium connecting several nodes. Data can be transferred via serial or parallel circuits, that is, via electrical or fiber optic conductors .

Bus connector

Physical connection between the bus node and the bus cable.

Bus node

This is a device that can send, receive or amplify data via the bus. It can be a DP master, DP slave, RS 485 repeater, active star coupler etc.

Bus segment

A bus segment is a self-contained section of a serial bus system. Bus segments are linked to one another using repeaters in PROFIBUS DP, for example.

Chassis ground

Chassis ground includes all the interconnected inactive parts of equipment that must not carry a hazardous voltage even in the event of a fault.

Clock memory

Memory bit which can be used to generate clock pulses in the user program (1 memory byte).

Note

Make sure that the clock memory byte is not overwritten in the user program.

Code block

A SIMATIC S7 logic block contains elements of the *STEP 7* user program. (in contrast to a DB: this contains only data.)

See also Data block

Compression

The programming device online function "Compress" is used to rearrange all valid blocks in CPU RAM in one continuous area of user memory, starting at the lowest address. This eliminates fragmentation which occurs when blocks are deleted or edited.

Configuration

Assignment of modules to slots and (for example with electronic modules) addresses.

Consistent data

Data which are related in their contents and not to be separated are referred to as consistent data.

For example, the values of analog modules must always be handled as a whole, that is, the value of an analog module must not be corrupted as a result of read access at two different points of time.

Counters

Counters are part of CPU system memory. The content of "Counter cells" can be modified by *STEP 7* instructions (for example, up/down count.)

See also System memory

CPU

Central processing unit = CPU of the S7 automation system with a control and arithmetic unit, memory, operating system, and interface for programming device.

Cycle control point

The cycle control point is the section of the CPU program processing in which the process image is updated.

Cycle time

The cycle time represents the time a CPU requires for one execution of the user program.

See also User program

See also CPU

Cyclic interrupt

→ *Interrupt, cyclic interrupt*

Data block

Data blocks (DB) are data areas in the user program which contain user data. There are shared data blocks which can be accessed by all code blocks, and instance data blocks which are assigned to a specific FB call.

Data exchange broadcast

→ *Direct data exchange*

Data exchange traffic

→ *Direct data exchange*

Data set routing

Functionality of a module with several network connections.
Modules that support this function are able to pass on data of an engineering system (for example parameter data generated by SIMATIC PDM) from a subnetwork to a field device at the PROFIBUS DP.

Data, static

Static data can only be used within a function block. These data are saved in an instance data block that belongs to a function block. Data stored in an instance data block are retained until the next function block call.

Data, temporary

Temporary data is the local data of a block. It is stored in the L-stack when the block is executed. After the block has been processed, this data is no longer available.

Diagnostic interrupt

Modules capable of diagnostics operations report detected system errors to the CPU by means of diagnostic interrupts.

See also CPU

Diagnostics

→ *System diagnostics*

Diagnostics buffer

The diagnostics buffer represents a buffered memory area in the CPU. It stores diagnostic events in the order of their occurrence.

Direct data exchange

Direct data exchange is a special communication relationship between PROFIBUS DP nodes. The direct data exchange is characterized by PROFIBUS DP nodes which "listen" on the bus and know which data a DP slave returns to its DP master.

Distributed I/O systems

I/O systems that are not integrated into the central controller, but rather at distributed locations a long distance from the CPU, such as:

- ET 200M, ET 200L, ET 200S, ET 200pro
- DP/AS-I Link
- S5-95U with PROFIBUS DP slave interface
- Further DP slaves supplied by Siemens or other vendors.

The distributed I/O systems are connected to the DP master via PROFIBUS DP.

DP master

→ *Master*

DP master

A master that complies with the IEC 61784-1:2002 Ed1 CP 3/1 standard is known as a DP master.

DP slave

→ *Slave*

DP slave

A slave running on the PROFIBUS using the PROFIBUS DP protocol in compliance with IEC 61784-1:2002 Ed1 CP 3/1 is known as a DP slave.

DP Standard

Bus protocol of the ET 200 distributed I/O system to IEC 61784-1:2002 Ed1 CP 3/1.

DPV1

The designation DPV1 means extension of the functionality of the acyclical services (to include new interrupts, for example) provided by the DP protocol. The DPV1 functionality is an integral part of the IEC 61784-1:2002 Ed1 CP 3/1 standard.

Electrically isolated

Electrically isolated I/O modules are isolated from the reference potentials of the control and load circuits by means of an optocoupler, relay contact or transformer circuit, for example. I/O circuits may be connected to the same potential.

Electronic modules

Electronic modules form the interface between the process and the automation system. There are

- digital input and output modules
- analog input and output modules
- Technology modules

Equipotential bonding

Electrical connection (equipotential bonding conductor) that keeps electrical equipment and extraneous conductive objects to the same or almost the same potential in order to prevent disturbing or dangerous voltages between those objects.

Error display

One of the possible reactions of the operating system to a runtime error is to output an error message. Further reactions: Error reaction in the user program, CPU in STOP.

See also Runtime error

See also Error reaction

Error handling via OB

After the operating system has detected a specific error (e.g. an access error with *STEP 7*), it calls a dedicated organization block (error OB) in which the subsequent behavior of the CPU can be defined.

Error response

Reaction to a runtime error. Reactions of the operating system: It sets the automation system to STOP, calls an OB in which the user can program a reaction or display the error.

See also Runtime error

ET 200

The ET 200 distributed I/O system with PROFIBUS DP or PROFINET IO allows the connection of distributed I/Os to a CPU via a DP master or IO controller. ET 200 is characterized by high-speed reaction times, because of a minimum data transfer volume (bytes.)

The ET 200 is based on IEC 61784-1:2002 Ed1 CP 3/1.d standard.

The ET 200 works on the master / slave principle or controller / device principle.
The DP masters are, for example, the IM 308-C master connection or the IM 151-7 CPU interface module with DP master module. An IO controller could be, for example, the IM 151-8 PN/DP CPU interface module.

DP slaves / IO devices could be the distributed I/Os ET 200M, ET 200L, ET 200S, ET 200pro or DP slaves / IO devices from Siemens or other vendors.

External lightning protection

External plant components at which galvanic coupling of lightning surges is excluded. Corresponds with lightning protection zone 0_A and 0_B.

FB

→ *Function block*

FC

→ *Function*

Flash EPROM

FEPROMs can retain data in the event of power loss, same as electrically erasable EEPROMs. However, they can be erased within a considerably shorter time (FEPROM = Flash Erasable Programmable Read Only Memory). They are used on SIMATIC Micro memory cards.

FORCE

The Force function can be used to assign the variables of a user program or CPU (also: inputs and outputs) constant values.

Note in this connection also the restrictions in section *Overview: Debugging functions* in section *Debugging functions, diagnostics and troubleshooting*.

FREEZE

Control command a DP master may broadcast to a group of DP slaves.

When it receives a FREEZE command, the slave freezes its current input status and outputs its data cyclically to the DP master.

The DP slave freezes its input status again after each new FREEZE command.

The DP slave does not resume the transfer input data to the DP master until the DP master has sent the UNFREEZE control command.

Function

According to IEC 1131-3, a function (FC) is a code block without static data. A function allows transfer of parameters in user program. Functions are therefore suitable for programming frequently occurring complex functions, e.g. calculations.

Function block

According to IEC 1131-3, a function block (FB) is a code block with static data. An FB allows the user program to pass parameters. Function blocks are therefore suitable for programming complex functions, e.g., closed-loop controls, mode selections, which are repeated frequently.

Functional ground

Grounding which has the sole purpose of safeguarding the intended function of electrical equipment. With functional grounding you short-circuit interference voltage which would otherwise have an unacceptable impact on equipment.

Ground

The conductive earth whose electrical potential can be set equal to zero at any point.

Ground potential may be different from zero in the area of grounding electrodes. The term reference ground is frequently used to describe this situation.

Ground-free

Having no direct electrical connection to ground

Grounding

Grounding means, to connect an electrically conductive component via an equipotential grounding system to a grounding electrode (one or more conductive components with highly conductive contact to earth).

GSD file

All slave-specific properties are stored in a GSD file (device master data file). The format of the GSD file is available in the IEC 61784-1:2002 Ed1 CP 3/1 standard.

Hot-swapping

The removal and insertion of modules during the operation of the ET 200S.

IM

Interface module: The interface module combines the ET 200S with the DP master and prepares the data for the electronic modules.
In an ET 200S with IM151-7 CPU, the IM 151-7 CPU interface module is itself the DP master (together with the DP master module).

Instance data block

The *STEP 7* user program assigns an automatically-generated DB to every call of a function block. The instance data block stores the values of input, output and in/out parameters, as well as local block data.

Interface, MPI-compatible

→ *MPI*

Internal lightning protection

Shielding of buildings, rooms or devices Corresponds with lightning protection zone 1, 2 or 3.

Interrupt

The operating system of an S7 CPU can distinguish between different priority classes that control how the user program is executed. These priority classes include interrupts, e.g. process interrupts. When an interrupt is triggered, the operating system automatically calls an assigned OB. In this OB the user can program the desired response (e.g. in an FB).

See also Operating system

Interrupt, cyclic interrupt

A cyclic interrupt is generated periodically by the CPU in a configurable time pattern. A corresponding OB will be processed.

See also Organization Block

Interrupt, delay

The delay interrupt belongs to one of the priority classes in SIMATIC S7 program processing. It is generated on expiration of a time started in the user program. A corresponding OB will be processed.

Interrupt, delay

→ *Interrupt, delay*

Interrupt, diagnostic

→ *Diagnostic interrupt*

Interrupt, process

→ *Process interrupt*

Interrupt, status

An status interrupt can be generated by a DPV1 slave. At the DPV1 master the receipt of the interrupt causes the OB 55 to be called.

For detailed information on OB 55, refer to the Reference Manual *System Software for S7-300/400*.

Interrupt, time-of-day

The time-of-day interrupt is one of the priority classes in SIMATIC S7 program processing. It is generated at a specific date (or daily) and time-of-day (e.g. 9:50 or hourly, or every minute). A corresponding OB will be processed.

Interrupt, update

An update interrupt can be generated by a DPV1 slave. At the DPV1 master the receipt of the interrupt causes the OB 56 to be called.

For detailed information on OB 56, refer to the Reference Manual *System Software for S7-300/400*.

Interrupt, vendor-specific

A manufacturer-specific interrupt can be generated by a DPV1 slave. At the DPV1 master the receipt of the interrupt causes the OB 57 to be called.

For detailed information on OB 57, refer to the Reference Manual *System Software for S7-300/400*.

Isochronous mode

The process data, transmission cycles via PROFIBUS DP, and the user program are synchronized in order to achieve ultimate deterministic. The input and output data of distributed I/O devices in the system are detected and output simultaneously. The isochronous PROFIBUS DP cycle form the corresponding clock generator.

Load memory

The load memory contains objects generated by the programming device. For the IM 151-7 CPU interface module, it takes the form of a plug-in SIMATIC Micro Memory Card with various memory sizes. A SIMATIC Micro Memory Card must be inserted to use the IM 151-7 CPU interface module.

Load power supply

Power supply for the load voltage for the power modules

Local data

→ *Data, temporary*

Master

When a master is in possession of the token, it can send data to other nodes and request data from other nodes (= active node). The DP masters are, for example, the CPU 315-2 DP or the IM 151-7 CPU interface module with DP master module.

Master

→ *Slave*

Micro Memory Card

→ *SIMATIC Micro Memory Card*

Module parameters

Module parameters are values which can be used to configure module behavior. There are two different types of parameter: static and dynamic.

MPI

The multipoint interface (MPI) represents the programming device interface of SIMATIC S7. It enables multiple nodes (PGs, text-based displays, OPs) to be operated simultaneously by one or more CPUs. Each device is identified by its unique (MPI address) address.

MPI address

→ *MPI*

Nesting depth

A block can be called from another by means of a block call. Nesting depth is referred to as the number of simultaneously called code blocks.

See also Code Blocks

Network

A network consists of one or more interconnected subnets with any number of nodes. Several networks can exist alongside each other.

Non-isolated

The reference potentials of the control and load circuit of non-isolated I/O modules are electrically interconnected.

OB

→ *Organization blocks*

OB priority

The CPU operating system distinguishes between different priority classes, for example, cyclic program execution or process interrupt-controlled program processing. Each priority class is assigned organization blocks (OBs) in which the S7 user can program a response. The OBs are assigned different default priority classes. These determine the order in which OBs are executed or interrupt each other when they appear simultaneously.

See also Operating system

See also Organization block

Operating state

SIMATIC S7 automation systems know the following operating states: STOP, STARTUP, RUN.

See also STARTUP, RUN

Operating system

The CPU OS organizes all functions and processes of the CPU which are not associated to a specific control task.

Operating system

→ *CPU*

Organization blocks

Organization blocks (OBs) form the interface between CPU operating system and the user program. The order in which the user program is executed is defined in the organization blocks.

Parameter

- Variable of a *STEP 7* code block
- Variable used to set the behavior of a module (one or more per module). All modules have a suitable basic factory setting which can be customized in *STEP 7*.
  There are static and dynamic parameters.

See also static parameters

See also dynamic parameters

Parameter assignment

Refers to the transfer of parameters from the DP master to the DP slave.

Parameters, dynamic

In contrast to static parameters, you can change dynamic module parameters in runtime by calling an SFC in the user program, e.g. limit values for an analog input module.

Parameters, static

In contrast to dynamic parameters, static parameters of modules cannot be changed by the user program. You can only modify these parameters by editing your configuration in *STEP 7*, for example, by modifying the input delay parameters of a digital input module.

PELV

Protective Extra Low Voltage = extra low voltage with safe isolation

PG

→ *Programming device*

PLC

→ *CPU*

PLC

Programmable controllers (PLCs) are electronic controllers whose function is saved as a program in the control unit. Therefore, the configuration and wiring of the unit does not depend on the PLC function. The programmable logic PLC has the structure of a computer; it consists of a CPU with memory, I/O modules and an internal bus system. The I/O and the programming language are oriented to control engineering needs.

PLC

→ *PLC*

PNO

→ *PROFIBUS International*

Priority class

The S7 CPU operating system provides up to 26 priority classes (or "Program execution levels"). Specific OBs are assigned to these classes. The priority classes determine which OBs interrupt other OBs. Multiple OBs of the same priority class do not interrupt each other. In this case, they are executed sequentially.

Process image

The process image is part of CPU system memory. At the start of cyclic program execution, the signal states at the input modules are written to the process image of the inputs. At the end of cyclic program execution, the signal status of the process image of the outputs is transferred to the output modules.

See also System memory

Process interrupt

A process interrupt is triggered by interrupt-triggering modules as a result of a specific event in the process. The process interrupt is reported to the CPU. The assigned organization block will be processed according to interrupt priority.

See also Organization Block

Product version

The product version identifies differences between products which have the same order number. The product version is incremented when forward-compatible functions are enhanced, after production-related modifications (use of new parts/components) and for bug fixes.

PROFIBUS

PROcess FIeld BUS, German process field bus standard specified in IEC 61784-1:2002 Ed1 CP 3/1. It specifies functional, electrical and mechanical properties for a bit-serial field bus system.
From the perspective of the user program, the distributed I/O is just as sophisticated as the centralized I/O.

PROFIBUS is available with the protocols DP (= Distributed Peripherals), FMS (= Fieldbus Message Specification), PA (= Process Automation), or TF (= Technological Functions.)

PROFIBUS address

A node must be assigned a unique PROFIBUS address in order to allow its identification on PROFIBUS.

The PC/Programming device is assigned PROFIBUS address "0."

The PROFIBUS addresses 1 to 125 may be used for the ET 200S distributed I/O system.

PROFIBUS DP

→ *PROFIBUS*

PROFIBUS International

Technical committee dedicated to the definition and development of the PROFIBUS standard.

Also known as the PROFIBUS User Organization membership corporation (PNO.)

Homepage: http://www.profibus.com

Programming device

Basically speaking, PGs are compact and portable PCs which are suitable for industrial applications. They are identified by a special hardware and software for programmable logic controllers.

Publisher

→ *Direct data exchange*

Publisher

The publisher is a sender in the direct data exchange.

RAM

RAM (Random Access Memory) is a semiconductor read/write memory.

Reference ground

→ *Ground*

Reference potential

Reference potential for the evaluation / measuring of the voltages of participating circuits.

Restart

When the IM 151-7 CPU interface module starts up (for example, after changing the mode selector switch from STOP to RUN or after a POWER ON), organization block OB 100 (warm restart) is executed before cyclic program processing (OB 1). On restart, the input process image is read in and the *STEP 7* user program is executed, starting at the first instruction in OB1.

Retentive memory

A memory area is considered retentive if its contents are retained even after a power loss and transitions from STOP to RUN. The non-retentive area of bit memory, timers and counters is reset following a power failure and a transition from the STOP mode to the RUN mode.

Retentive can be the:

- Bit memory
- S7 timers
- S7 counters
- Data areas

Runtime error

Errors occurred in the PLC (that is, not in the process itself) during user program execution.

Segment

→ *Bus segment*

SELV

Safety Extra Low Voltage

SFB

→ *System function block*

SFC

→ *System function*

SIMATIC

The term denotes Siemens products and systems for industrial automation.

SIMATIC Micro Memory Card

SIMATIC Micro Memory Cards are storage media for the IM 151-7 CPU interface module.

Slave

→ *Master*

Slave

A slave can only exchange data after being requested to by the master.

STARTUP

A STARTUP routine is executed at the transition from STOP to RUN mode. Can be triggered by means of the mode selector switch, or after power on, or by an operator action on the programming device. A restart has been carried out on the IM 151-7 CPU interface module.

See also Mode selector switch

See also Restart

STEP 7

Engineering system. Contains programming software for the creation of user programs for SIMATIC S7 controllers.

Subscriber

→ *Direct data exchange*

Subscriber

The subscriber is a recipient in the direct data exchange.

Substitute value

Substitute values are configurable values which output modules transfer to the process when the CPU switches to STOP mode.

In the event of an I/O access error, a substitute value can be written to the accumulator instead of the input value which could not be read (SFC 44).

SYNC

Control command a DP master may broadcast to a group of DP slaves.

With the SYNC control command the DP master causes the DP slave to freeze the statuses of the outputs at the current value. The DP slave stores the output data contained in the next frame, but does not change the state of its outputs.

After each new SYNC control command, the DP slave sets the outputs it has saved as output data. The outputs are not updated cyclically again until the DP master has sent a UNSYNC control command.

System diagnostics

System diagnostics refers to the detection, evaluation, and signaling of errors that occur within the PLC, for example programming errors or module failures. System errors can be indicated by LEDs 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 when necessary in the *STEP 7* user program.

System function block

A system function block (SFB) is a function block integrated in the operating system of the CPU that can be called when necessary in the *STEP 7* user program.

System memory

System memory is an integrated RAM memory in the CPU. System memory contains the address areas (e.g. timers, counters, bit memory) and data areas that are required internally by the operating system (for example, communication buffers).

See also Operating system

System status list

The system status list contains data that describes the current status of an ET 200S with IM 151-7 CPU interface module. You can always use this list to obtain an overview of:

- the configuration of the ET 200S
- The current CPU parameter assignments and configurable electronic modules
- The current statuses and processes in the CPU and the configurable electronic modules

Terminating module

The ET 200S distributed IO system is completed by the terminating module. If you have not inserted a terminating module, the ET 200S is not ready for operation.

Terminating resistor

The terminating resistor is used to avoid reflections on data links.

Time-of-day interrupt

→ *Interrupt, time-of-day*

Timer

→ *Timers*

Timers

Timers are part of CPU system memory. The content of timer cells is automatically updated by the operating system, asynchronously to the user program. *STEP 7* instructions are used to define the precise function of the timer cell (for example, switch-on delay) and to initiate their execution (for example, start).

See also System memory

Token

Allows access to the PROFIBUS DP for a limited time.

Topology

Structure of a network. Common structures include:

- Bus topology
- Ring topology
- Star topology
- Tree topology

User program

In SIMATIC, a distinction is made between the operating system of the CPU and user programs. The user program contains all instructions, declarations and data for signal processing required to control a plant or a process. It is assigned to a programmable module (for example CPU) and can be structured in smaller units (blocks).

User program

→ *Operating system*

User program

→ *STEP 7*

Voltage group

A group of electronic modules supplied by one power module.

Work memory

The work memory is integrated in the CPU and cannot be extended. It is used to run the code and process user program data. Programs only run in work memory and system memory.

See also CPU

Index

L

M

N

O

T

U

V

W