SIEMENS

# SIMATIC

# S7-300 Programmable Controller CPU Specifications, CPUs 312C to 314C-2DP/PtP

**Reference Manual**

Contents




This manual is part of the documentation package with the order number:
**6ES7398-8FA10-8BA0**

**Edition 10/2001**
**A5E00105475-01**

## Safety Guidelines

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

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

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

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

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

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

## Qualified Personnel

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

## Correct Usage

Note the following:

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

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

## Trademarks

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

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



**Disclaimer of Liability**

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

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

# Contents

## Figures

## Tables

# Preface 1

## Purpose of this manual

This manual provides an overview of the S7-300 CPUs 312C to 314C-2 PtP/DP.

Here you can look up information on operation, functions and technical data of the CPUs.

## Basic knowledge required

For comprehension of this manual you require a general knowledge of automation control engineering. You should also be familiar with basic STEP 7 software, introduced in the Manual *Programming with STEP 7 V5.1*.

## Range of validity of this manual

This manual applies to the CPUs listed below, operated with the following hardware and software versions:

| CPU | Order No. | As of Version | |
|---|---|---|---|
| | | Firmware | Hardware |
| CPU 312C | 6ES7 312-5BD00-0AB0 | V1.0.0 | 01 |
| CPU 313C | 6ES7 313-5BE00-0AB0 | V1.0.0 | 01 |
| CPU 313C-2 PtP | 6ES7 313-6BE00-0AB0 | V1.0.0 | 01 |
| CPU 313C-2 DP | 6ES7 313-6CE00-0AB0 | V1.0.0 | 01 |
| CPU 314C-2 PtP | 6ES7 314-6BF00-0AB0 | V1.0.0 | 01 |
| CPU 314C-2 DP | 6ES7 314-6CF00-0AB0 | V1.0.0 | 01 |

This manual contains the description of all modules valid at the time this edition was released. For new modules or newer versions of modules, we reserve the option to add to the manual a product information containing the current information on this module.

## Approvals, Standards, Releases

The SIMATIC S7-300 product series is approved by:

- Underwriters Laboratories, Inc.: UL 508 (Industrial Control Equipment)
- Canadian Standards Association: CSA C22.2 No. 142, (Process Control Equipment)
- Factory Mutual Research: Approval Standard Class Number 3611

## CE label

The SIMATIC S7-300 product series conforms with the requirements and safety specifications of following EC guidelines:

- EC Guideline 73/23/EEC Low-Voltage Guideline
- EC Guideline 89/336/EEC EMC Guidelines

## C-Tick-Mark

The SIMATIC S7-300 product series is compliant with AS/NZS 2064 (Australia).

## Standards

The SIMATIC S7-300 product series is compliant with the requirements and criteria of IEC 61131-2.

## Its place in the world of documentation

This manual forms part of the S7-300 documentation package.

| | |
|---|---|
| **Reference manual "CPU Data"**<br>"CPU Data CPU 312 IFM to 318-2 DP"<br>→ "CPU Data CPU 312 C to314C-2 PtP/DP" | Description of the operation, the functions and the technical data of the CPU |
| **Manual "Technological Functions"**<br>Manual<br>Examples | Description of the individual technological functions:<br>- Positioning<br>-Count<br>-Point-to-point connection<br>-Control<br>The CD contains examples on the technological functions. |
| **Installation Manual**<br>Manual | Configuration, installation, wiring, network and commissioning descriptions of a S7-300 |
| **Reference manual "Module data"**<br>Manual | Functions descriptions and technical data of the signal modules, power supply modules and the interface modules |
| **Operating lists**<br>"CPU 312 IFM, 314 IFM, 313, 315, 315-2 DP, 316-2 DP, 318-2 DP"<br>"CPUs 312 C to 314C-2 PtP/DP" | List of stored instructions of the CPUs and their execution times.l<br>List of executable blocks (OBs/SFCs/SFBs) and their execution times. . |
| **Getting Started**<br>"CPU 31xC: Positioning with analog output"<br>"CPU 31xC: Positioning with digital output"<br>"CPU 31xC: Count"<br>"CPU 31xC: Point-to-point connection"<br>"CPU 31xC: Control "<br>"CPU 31xC"<br>"S7-300" | Getting Starteds take you through each commissioning step up to a functioning application by running through a concrete example. |

Read this manual

Figure 1-1 Information environment of S7-300

You also require the following supplementary manuals to this documentation package:

| | |
|---|---|
| Manual<br>"Integrated function CPU 312 IFM/314 IFM"<br><br>Manual<br>order numer: 6ES7398-8CA00-8BA0 | Description of the technological functions of the CPUs 312 IFM/314 IFM. |
| Reference manual "system software for S7-300/400 system and standard functions"<br><br>Reference manual<br>Part of the STEP 7 documentation package<br>with the order number: 6ES7810-4CA05-8BR0 | Description of the SFCs, SFBs, OBs and CPUs.<br>You can also find the description in the STEP 7 Online Help |

Figure 1-2 Additional Documentation

## Further Support

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

http://www.ad.siemens.de/partner

## Training Center

Newcomers are welcomed to take part in our S7-300 PLC courses. Please contact your regional Training Center or the central Training Center in D-90327 Nuremberg, Germany.

Phone: +49 (911) 895-3200

http://www.sitrain.com

## SIMATIC Documentation on the Internet

You can find free-of-charge documentation on the Internet under:

http://www.ad.siemens.de/support

There, you can use our Knowledge Manager to quickly find your required documentation. Our Internet conference forum is available to receive your questions and suggestions relating to documentation.

## SIMATIC Customer Support Hotline

Worldwide available 24 hours:

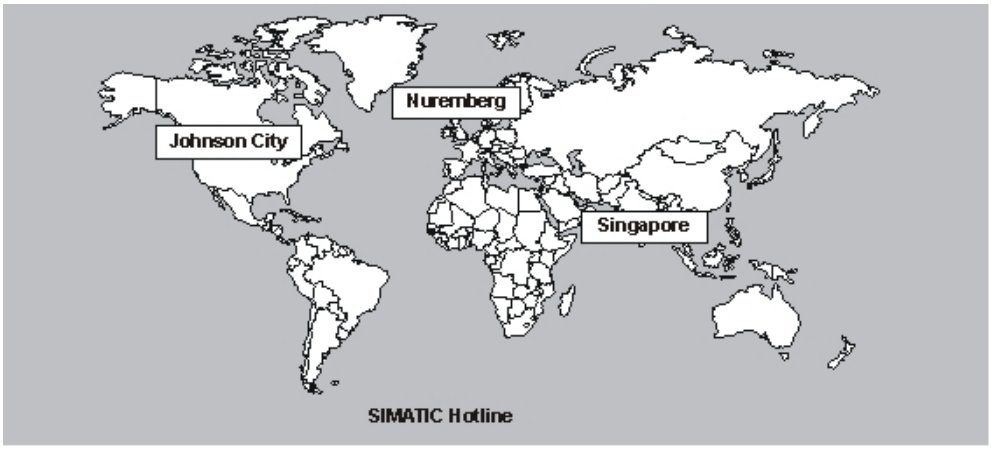



Figure 1-3 SIMATIC Customer Support Hotline

| | | |
|---|---|---|
| **Worldwide** (Nuremberg)<br>**Technical Support**<br>(Free Contact)<br>Local time: Mo.-Fr. 7:00 to 17:00<br>Phone: +49 (180) 5050 222<br>Fax: +49 (180) 5050 223<br>E-mail: techsupport@ad.siemens.de<br>GMT: +1:00 | **Worldwide** (Nuremberg)<br>**Technical Support**<br>(charged, only with SIMATIC Card)<br>Local time: Mo.-Fr. 0:00 to 24:00<br>Phone: 49 (911) 895-7777<br>Phone: 49 (911) 895-7001<br>GMT: +1:00 | |
| **Europe / Africa** (Nuremberg)<br>**Authorization**<br>Local time: Mo.-Fr. 7:00 to 17:00<br>Phone: 49 (911) 895-7200<br>Phone: 49 (911) 895-7201<br>E-mail: authorization@nbgm.siemens.de<br>GMT: +1:00 | **America** (Johnson City)<br>**Technical Support and Authorization**<br>Local time: Mo.-Fr. 8:00 to 19:00<br>Phone: +1 423 262 -2522<br>Fax: +1 423 262 -2289<br>E-mail: simatic.hotline@sea.siemens.com<br>GMT: -5:00 | **Asia / Australia** (Singapore)<br>**Technical Support and Authorization**<br>Local time: Mo.-Fr. 8:30 to 17:30<br>Phone: +65 740 -7000<br>Fax: +65 740 -7001<br>E-mail: simatic.hotline@sea.siemens.com.sg<br>GMT: +8:00 |
| Languages spoken at the SIMATIC Hotlines are German and English. Additional languages spoken at our Authorization Hotline are French, Italian and Spanish. | | |

## Service & Support on the Internet

Our documentation is supplemented by our complete know-how available to you Online on the Internet.

http://www.ad.siemens.de/support

There you will find:

- Up-to-date product information (News), FAQs (Frequently Asked Questions), Downloads, Tips and Tricks.
- Our Newsletter always offers you the latest information on your products.
- The Knowledge Manager finds the right documents for you.
- Users and specialists from all over the world share their experience.
- You can find your local service partner for Automation & Drives in our Service Partner database.
- Information relating to on-site service, repairs, spare parts and lots more is available to you in the Service section.

# Quick Guide: selecting and arranging 2

### In this Quick Guide ...

you can find - based on the conditions and requirements for your S7-300 system application - a 'signpost' leading you through the S7-300 documentation.

### The place of S7-300 in S7 Technology

Table 2-1 S7-300 in S7 Technology

| S7-200 | S7-300 | S7-400 | C7 |
|---|---|---|---|
| Fast and versatile Micro-PLC | Fast, high-performance and versatile Compact PLC (CPUs 31xC with integrated functionality) | Fast, high-performance, rugged and versatile Power PLC with high-performance communication capabilities | Compact and compatible Complete PLC |
| - | Available Documentation  | - | - |

## Selection and Configuration

Table 2-2 Ambient influence on the PLC

| Information on ... | is available in ... |
|---|---|
| What provisions do I have to make for PLC installation space? | Chapter *Configuring; Mounting dimensions of modules* and *mounting; Mounting profile rails, in the installation manual* |
| How do environmental conditions influence the PLC? | *Appendix of the Installation Manual* |

Table 2-3 Potential isolation

| Information on ... | is available in ... |
|---|---|
| Which modules can I use if it is required to isolate the potential of specific sensors/actuators? | Chapter *Configuring; Electrical assembly, Protective measures and Grounding, in the Installation Manual*<br>Reference Manual *Module Specifications* |
| When is it required to isolate the potential of specific modules?<br>How do I wire that? | Chapter *Configuring; Electrical assembly, Protective measures and Grounding, in the Installation Manual*<br>Chapter *Wiring, in the Installation Manual* |
| When is it required to isolate the potential of specific stations?<br>How do I wire that? | Chapter *Configuring; Configuring subnets, in the Installation Manual*<br>Chapter *Wiring, in the Installation Manual* |

Table 2-4 Communication between sensors/actuators and the PLC

| Information on ... | is available in ... |
|---|---|
| Which module is suitable for my sensor/actuator? | for CPU: Reference Manual *CPU Data*<br>for signal modules: Reference Manual *Module Data* |
| How many sensors/actuators can I connect to the module? | for CPU: Reference Manual *CPU Data*<br>for signal modules: Reference Manual *Module Data* |
| How do I wire the sensors/actuators to the PLC using front connectors? | Chapter *Wiring; Wiring front connectors, in the Installation Manual* |
| When do I require expansion modules (EG), and how are they connected? | Chapter *Configuring, optional expansions and networking, in the Installation Manual* |
| How do I mount modules on module racks/profile rails? | Chapter *Mounting; Mounting modules on profile rails, in the Installation Manual* |

Table 2-5 Application of central and distributed I/O

| Information on ... | is available in ... |
|---|---|
| Which range of modules do I want to use? | for central I/O / expansion modules (EGs): Reference Manual *Module data*<br>for distributed I/O / PROFIBUS-DP: Manual of the respective peripheral device, e.g. *Manual ET 200B* |

Table 2-6 Configuration consisting of the Central Unit (CU) and Expansion Modules (EMs)

| Information on ... | is available in ... |
|---|---|
| Which mounting racks / profile rails are suited best for my application? | Chapter *Configuring, in the Installation Manual* |
| Which Interface modules (IM) do I need for connecting EMs to the CPU? | Chapter *Configuring, Arranging modules on multiple module racks, in the Installation Manual* |
| How do I configure the power supply (PS) performance? | Chapter *Configuring, in the Installation Manual* |

Table 2-7 CPU performance

| Information on ... | is available in ... |
|---|---|
| Which is the most adequate memory setup for my application? | Reference manual *CPU Data* |
| How do I install and remove Micro Memory Cards? | Chapter *Commissioning; Removing/Installing Micro Memory Cards, in the Installation Manual* |
| Which CPU meets my requirements on performance ? | *Instruction List*; Reference Manual *CPU Data* |
| How long are CPU response and processing times? | Reference manual *CPU Data* |
| Which technological functions are implemented? | Manual *Technological functions* |
| How can I use these functions? | Manual *Technological functions* |

Table 2-8 Communications

| Information on ... | is available in ... |
|---|---|
| Which principles do I have to take into account? | Manual *Communication with SIMATIC* |
| Which options and resources does the CPU offer? | Reference manual *CPU Data* |
| How do I optimize communication with the help of communication processors (CPs)? | the respective device Manual |
| Which communication network is most suitable for my application? | Chapter *Configuring; Configuring subnets, in the Installation Manual*<br>Manual *Communication with SIMATIC* |
| How do I network the specific modules? | Chapter *Configuring* and *wiring, in the Installation Manual* |

Table 2-9 Software

| Information on ... | is available in ... |
|---|---|
| Which software do I require for my S7-300 system? | Chapter *Technical data*; Reference Manual *CPU data* |

Table 2-10 Supplementary features

| Information on ... | is available in ... |
|---|---|
| How do I implement operator control and monitoring?<br>(Human Machine Interface) | for text displays: the respective device manual<br>for operator panels: the respective device manual<br>for WinCC: the respective device manual |
| How can I integrate process control modules? | for PCS 7: the respective device manual |
| What are the options of redundant and fail-safe systems? | Manual *S7-400H - Redundant systems*; Manual *Fail-safe systems* |

## Cross-reference

Please note that you can also download respective current manuals from our public Internet URL. Details are found in the *Preface*.

# Structure and communication functions of a CPU 31xC

# 3

## 3.1 Control and Display Elements

### Elements

The figure below shows the control and display elements of a CPU 31xC. Arrangement and number of elements differ in some CPUs.

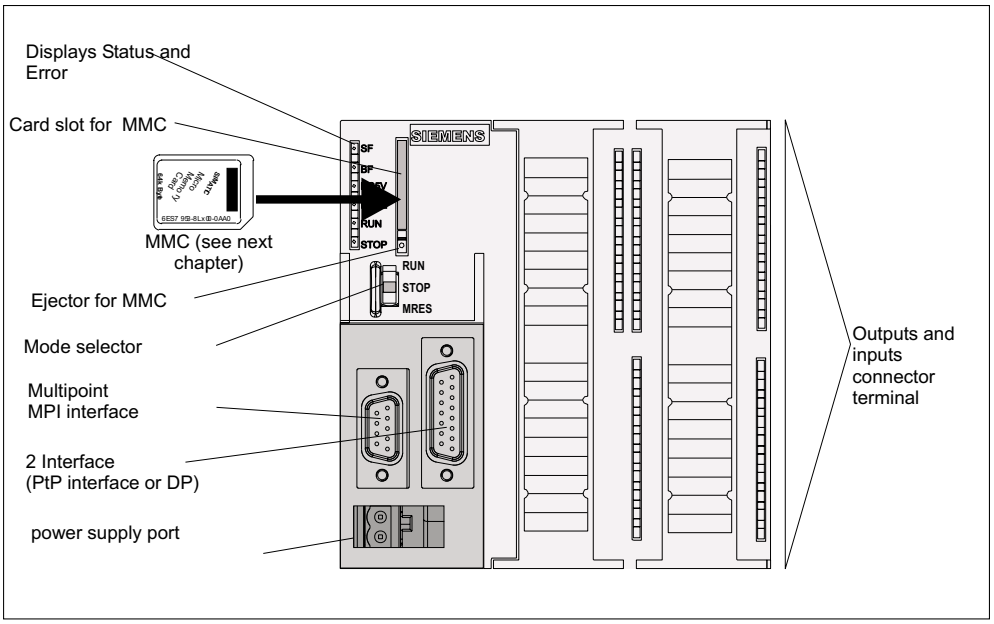



Figure 3-1 Elements and assembly of a CPU 31xC, for example, a CPU 314C-2 PtP

The figure below shows the digital/analog I/O integrated in the CPU.

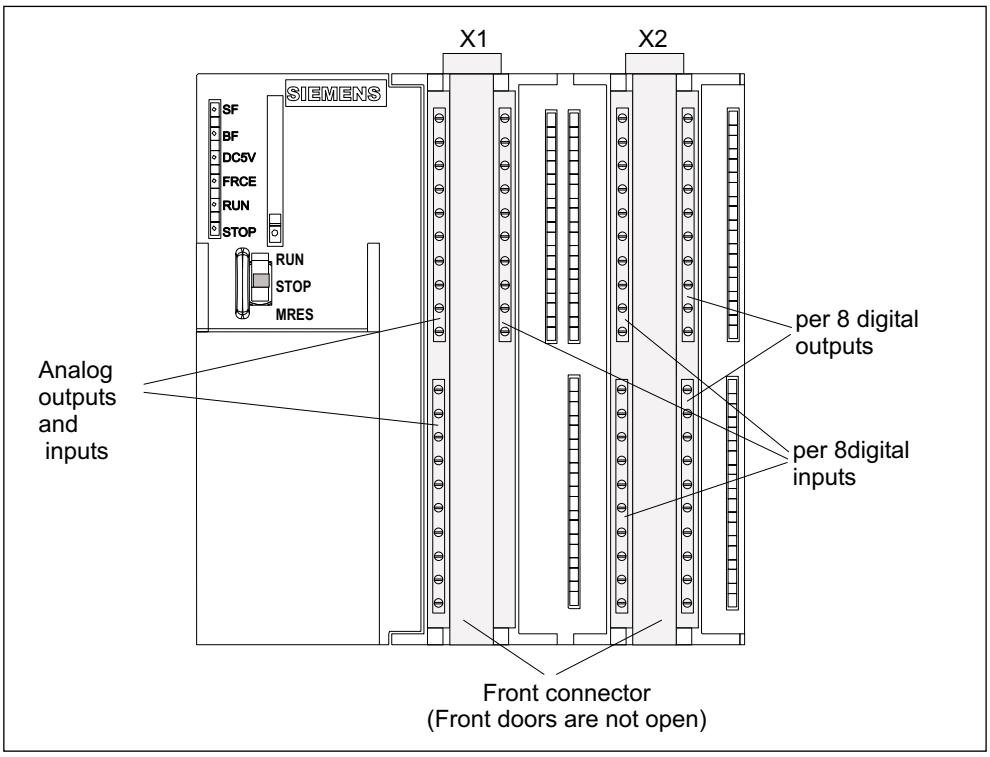



Figure 3-2 Integrated I/O of the CPU 31xC, for example, a CPU 314C-2 PtP

## Differences Between CPUs

Table 3-1 Differences Between CPUs

| Element | CPU 312C | CPU 313C | CPU 313C-2 DP | CPU 313C-2 PtP | CPU 314C-2 DP | CPU 314C-2 PtP |
|---|---|---|---|---|---|---|
| SIMATIC Micro Memory Card (always required for operation) | x | x | x | x | x | x |
| 9-pin MPI interface | x | x | x | x | x | x |
| 9-pin DP interface | – | – | x | – | x | – |
| 15-pin PtP interface | – | – | – | x | – | x |
| Digital inputs | 10 | 24 | 16 | 16 | 24 | 24 |
| Digital outputs | 6 | 16 | 16 | 16 | 16 | 16 |
| Analog inputs | – | 4 + 1 | – | – | 4 + 1 | 4 + 1 |
| Analog outputs | – | 2 | – | – | 2 | 2 |
| Technological functions | 2 counters | 3 counters | 3 counters | 3 counters | 4 counters<br>1 Channel for positioning | 4 counters<br>1 Channel for positioning |

## Status and Error Displays

The CPU is equipped with the following LED displays:

**CPU LEDs:**

| | | |
|---|---|---|
| SF | (red) | Hardware or software error |
| BF | (red) | Bus error (only CPU 313C-2 DP and 314C-2 DP) |
| DC5V | (green) | The 5 V power supply for the CPU and S7 300 bus is ok |
| FRCE | (yellow) | Force job is active |
| RUN | (green) | CPU is in RUN; LED blinks during restart at 2 Hz; in HOLD at 0.5 Hz |
| STOP | (yellow) | CPU is in STOP or HOLD or restart; LED blinks during memory reset request at 0.5 Hz, during memory reset at 2 Hz. |

Figure 3-3 Status and Error Displays

## Diagnostics

How you can use the LED displays is explained in the Installation Guide, Chapter *Testing functions, Diagnostics and Fault Elimination*.

## Slot for the SIMATIC Micro Memory Card (MMC)

SIMATIC Micro Memory Cards (MMCs) are used as memory submodule for CPUs 31xC. The MMC can be used as load memory or portable storage medium. The MMC **must** be inserted to operate the CPU, since CPUs 31xC are not equipped with an integrated load memory.

## Mode Selector Switch

You can use the mode selector switch to set the current CPU operating mode.The switch has 3 positions.

## Positions of the mode selector switch

The positions of the mode selector are explained in the order in which they appear on the CPU.

Table 3-2 Mode Selector Positions

| Position | Mode | Description |
|---|---|---|
| RUN | RUN mode | The CPU scans the user program. |
| STOP | STOP mode | The CPU does not scan user programs. |
| MRES | Memory Reset | Mode selector switch position with pushbutton function for resetting CPU memory. Memory reset requires a specific sequence of operation (refer to the Installation Guide, Chapter *Commissioning*). |

## Power Supply Connection

The CPU 31xC series are equipped with a 2-pole power supply socket. The plug with screw terminals is included in the delivery and already inserted into the CPU.

## Further Information

Further information on CPU operating modes is found in the *STEP 7 Online Help*.

For information on resetting memory per mode selector operation please refer to the Installation Guide, Chapter *Commissioning*.

Details on error / diagnostics evaluation per LEDs are found in your Installation Guide, Chapter *Testing functions, Diagnostics and Fault Elimination*. Information on MMC usage and memory conception are found in the same chapters.

## 3.2 SIMATIC Micro Memory Card (MMC)

### Memory submodule

SIMATIC Micro Memory Cards (MMCs) are used as memory submodule for CPUs 31xC. You can use MMCs as load memory or portable storage medium. It is always required for CPU operation The following data is stored on MMC:

- User programs (all function blocks)
- Archives and recipes
- Configuration data (STEP 7 projects)
- Operating system update and backup data

**Note**

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

### Properties

SIMATIC Micro Memory Cards ensure maintenance-free and redundant operation of CPUs 31xC. Details are found in Chapter *Memory conception*.

**Caution**

Data on a SIMATIC Micro Memory Card can be corrupted if you remove the card during write access. In this case you might have to insert the MMC memory in your PG to delete it, or you format the card in the CPU.
Never remove an MMC in RUN mode. Always remove it after power off or in CPU STOP state, that is, when the PG does not write access the card. Disconnect the communication lines if you cannot safely exclude active write access functions from the PG (e.g. load/delete function block).

### Service life of an MMC

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

1. The number of delete or programming steps
2. external influences such as ambient temperature.

At an ambient temperature of up to 60°C, the MMC has a service life of 10 years, with a maximum of 100,000 delete/write operations.

**Caution**

As a precaution against data loss, always make sure that the maximum number of delete/write operations is not exceeded.

## Insertable SIMATIC Micro Memory Cards

The following memory submodules are available:

Table 3-3 Available MMCs

| Type | Order numbers | Required for Firmware Update with ... |
|---|---|---|
| MMC 64k | 6ES7 953–8LF00–0AA0 | – |
| MMC 128k | 6ES7 953–8LG00–0AA0 | – |
| MMC 512k | 6ES7 953–8LJ00–0AA0 | – |
| MMC 2M | 6ES7 953–8LL00–0AA0 | CPU 312C / CPU 313C / CPU 31xC-2 PtP |
| MMC 4M | 6ES7 953–8LM00–0AA0 | CPU 31xC-2 DP |

## Formatting MMC when resetting memory

Some specific situations require of you to format the MMC:

- The module type is not an application module.
- The MMC is faulty, corrupted or has not been formatted.

MMC memory is marked invalid.

- The *The load application program* instruction was interrupted by Power Off (see special handling).
- *Write FEPROM* was interrupted as a result of Power Off (see special handling).
- Module memory evaluation error during memory reset.
- Error when formatting or formatting was not possible.

If one of these errors has occurred, the CPU prompts you to repeat memory reset, regardless of a previous memory reset. Card memory is retained until a special handling is carried out, except if Load Application Program/Write to FEPROM operations have been interrupted as a result of Power Off.

**Description of Special Handling:**

When the CPU requests a memory reset (slow flashing of the STOP LED), format it using the following switching sequence:

1. Toggle the switch to MRES position and hold it there until the LED stops flashing (until permanently lit after approx. 9 seconds).
2. Within the next three seconds, release the switch and toggle it once again to MRES position. The STOP LED now flashes to indicate the formatting process.

**Always maintain the specified sequence of operation. Otherwise, the MMC is not formatted, but rather returns to memory reset status.**

The MMC is formatted for specific reasons only (see above); not, for example, in case of a memory reset request after module replacement. In this case, switching to MRES merely initiates a normal memory reset, while existing MMC data is retained.

### Further details on MMCs ...

are found in your Installation Guide.

## 3.3 Interfaces

### MPI Interface

MPI (Multi Point Interface) represents the interface between the CPU and a PG/OP or for communication in an MPI Subnet. Every CPU is equipped with an RS485 **MPI Interface**.

Typical (default) transmission speed is 187.5 Kbps. You can also specify 19.2 Kbps for communication with an S7-200. Other transmission rates are not possible.

At the MPI interface the CPU automatically broadcasts its bus parameter configuration (e.g. the baud rate). A programming device, for example, can then automatically retrieve the correct parameters and connect to an MPI Subnet.

---

**Note**

In RUN mode, you may only connect PGs to the MPI Subnet.
Other stations (e.g.. OP, TP, ...) should not be connected to the MPI subnet while the PLC is in run mode. Otherwise, transferred data might be corrupted as a result interference or global data packages be lost.

---

### PROFIBUS-DP interface

The PROFIBUS-DP interface is mainly used to connect distributed I/O. PROFIBUS-DP allows you to create extensive subnets, at a maximum transmission speed of 12 Mbps.

At the MPI interface the CPU automatically broadcasts its bus parameter configuration (e.g. the baud rate). A programming device, for example, can then automatically retrieve the correct parameters and connect to a PROFIBUS subnet. In your configuration, you can disable this bus parameter broadcast.

CPUs 313C-2 DP and 314C-2 DP are equipped with a **PROFIBUS-DP interface** that you can configure as master or as slave.

## PtP interface

You can use the CPU's PtP interface to connect devices equipped with a serial port, e.g. barcode readers, printers, ..., (non-Siemens devices). Here, you can realize a transmission speed of up to 19.2 Kbps in full duplex mode (RS 422) and 38.4 Kbps in half duplex mode (RS 485). CPUs 313C-2 PtP and 314C-2 PtP are equipped with a **PtP interface** (Point to Point). The following PtP communication drivers are installed in the CPUs:

- ASCII driver
- 3964 (R) Protocol
- RK 512 (only CPU 314C-2 PtP)

## Which devices can I connect to which interface?

Table 3-4 Connectable Devices

| MPI | PROFIBUS-DP | PtP |
|---|---|---|
| • PG/PC<br>• OP/TP<br>• S7-300/400 with MPI interface<br>• S7-200 (19.2 Kbps only) | • DP Slaves<br>• DP Master<br>• Actuators/Sensors<br>• S7-300/400 with PROFIBUS-DP interface<br>• PG/PC<br>• OP/TP | • Devices equipped with a serial port, e.g. barcode readers, printers, etc. |

## Further Information

Further information on specific connections is found in the *Communication with SIMATIC* Manual.

Details on PtP communication is found in the *Technological Functions* Manual.

## 3.4 Realtime clock

### Properties and Functions

The table below shows the properties and functions of the real-time clock.

Table 3-5 Properties and functions of the real-time clock

| Characteristics | CPU 312C | CPU 313C / CPU 313C-2 / CPU 314C-2 |
|---|---|---|
| Type | Software clock | Hardware clock |
| Manufacturer setting | DT#1994-01-01-00:00:00 | DT#1994-01-01-00:00:00 |
| Backup | No | with integrated capacitor |
| Backup period | – | normally 6 weeks (at an ambient temperature of 40°C) |
| Behavior of Clock in POWER OFF Mode | After power is returned, the clock resumes operation with the TOD status at power off. | The clock continues to operate in POWER OFF mode. |
| Behavior of the clock on expiration of the backup period | – | After power is returned, the clock resumes operation with the TOD status at power off. |

### Information on ...

- Synchronization and correction factor:

  When you configure your CPU in *STEP 7*, you can customize functions such as synchronization via MPI interface and the correction factor. Refer to the *Online Help for STEP 7*.

- Setting, reading and programming the real-time clock:

  You can retrieve and set the TOD with the help of your PG (refer to the Manual *Programming with STEP 7*), or you program the respective SFCs in your application program (refer to the *System and Standard functions*) Reference Manual.

## 3.5 Communications

### Communication Services of the CPUs

Your selected communication service influences

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

User interface characteristics can be quite different (SFC, SFB, ...), depending on the hardware used (SIMATIC CPU, PC, ...).

The CPUs provide the following communication services:

Table 3-6 Communication Services of the CPUs

| Communication Service | Functionality | Establishing the S7 connection ... | via MPI | via DP | via PtP |
|---|---|---|---|---|---|
| PG communication | Start-up, test, diagnostics | via PG at the moment the service is called | x | x | – |
| OP communication | Operating and monitoring | via OP at POWER ON | x | x | – |
| S7 basic communication | Data exchange | is programmed via function blocks (SFC parameters) | x | – | – |
| S7 communication | Data exchange | CPU 31xC only as server; communication is established by the partner | x | – | – |
| Global Data Communication | cyclic data exchange (e.g. memory bits) | does not require an S7 connection | x | – | – |
| Routing of PG functions<br>(only CPU 31xC-2 DP) | e.g. testing, diagnostics extending over network limits | via PG at the moment the service is called | x | x | – |
| Point-to-point connection | Data exchange via serial interface | does not require an S7 connection | – | – | x |

### PG Communication

PG communication is used for data exchange between engineering stations (e.g. PG, PC) and communication-capable SIMATIC modules. This service is possible on MPI / PROFIBUS / Industrial Ethernet subnets. Transition between subnets is also supported. PG communication provides functions required for loading programs and configuration data, as well as for testing and evaluating diagnostic information. These functions are integrated in the operating system of SIMATIC S7 modules.

A CPU can maintain several simultaneous online connections to one or multiple PGs.

## OP Communication

OP communication is used for data exchange between operator stations (e.g. OP, TP) and communication-capable SIMATIC modules. This service is possible on MPI / PROFIBUS / Industrial Ethernet subnets.

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

A CPU can maintain several simultaneous connections to one or several operator panels.

## S7 Basic Communication

S7 Basic Communication is used for data exchange between S7 CPUs and the communication-capable SIMATIC modules of an S7 station (acknowledged data exchange). Data exchange takes place via non-configured S7 connections. The service can be used on an MPI subnet or for internal communication between the station and function modules (FM).

S7 basic communication provides functions required for data exchange. These functions are integrated in the operating system of 31xC CPUs.

The user can utilize this service via "System function" (SFC) user interface.

## S7 Communication

CPUs 31xC operate as server in S7 communication. The connection is always established by the communication partner. This service is possible on MPI / PROFIBUS / Industrial Ethernet subnets.

The operating system processes these services without explicit user interface.

## Global Data Communication

Global Data Communication is used for cyclic exchange of global data (e.g. I, Q, M) between SIMATIC S7 CPUs (data exchange with no acknowledgement). One CPU broadcasts the data to all CPUs on the MPI subnet. This function is integrated in the operating system of 31xC CPUs.

**Send and Receive Conditions**

For GD circuit communication, you should always maintain the following conditions:

- For the station sending a GD package:
  Reduction factor $_{\text{Sending station}}$ x Cycle time$_{\text{Sending station}}$ 60 ms
- For the station receiving a GD package:
  Reduction factor$_{\text{Receiving station}}$ x Cycle time$_{\text{Receiving station}}$
  < Reduction factor$_{\text{Sending station}}$ x Cycle time$_{\text{Sending station}}$

A GD package might be lost if you do not maintain these conditions. The reasons for this are:

- The performance of the "smallest" CPU in the GD circuit
- The sending/receiving stations exchange global data asynchronously

if you specify in *STEP 7* : "Send data after every CPU cycle" - with a short CPU cycle time (< 60 ms) - the operating system might overwrite the CPU's GD package before it is transmitted. Loss of global data is indicated in the status bar of the GD circuit, provided you have configured this feature in *STEP 7.*

**Reduction factor**

The reduction factor specifies GD communication cycle intervals. You can customize this reduction factor when you configure global data communication in *STEP 7* . For example, if you select a reduction factor of 7, global data is transferred at intervals of 7 cycles. This reduces CPU load.

**GD Resources**

The table below shows the GD resources of CPUs 31xC.

Table 3-7 GD resources of CPUs 31xC

| Parameters | CPU 31xC |
|---|---|
| Number of GD circuits per CPU | max. 4 |
| Number of Send GD packets per GD circuit | max. 1 |
| Number of Send GD packets of all GD circuits | max. 4 |
| Number of Receive GD packets per GD circuit | max. 1 |
| Number of Receive GD packets of all GD circuits | max. 4 |
| Data length per GD packet | max. 22 bytes |
| Consistency | max. 22 bytes |
| Reduction factor (default) | 1 (8) |

## Routing

A CPU 31xC-2 DP master configuration in *STEP 7* as of V 5.1 + SP 2 allows communication between the PG/PC and S7 stations across subnet boundaries (MPI interface / PROFIBUS-DP interface), for example, to download user programs or a hardware configuration, or to execute testing and commissioning routines.

## Point-to-Point Communication

PtP communication enables data exchange via serial interface. PtP communication can be used to interconnect automation devices, computers or other communication-capable non-Siemens systems. The communication partner's protocol can be adapted accordingly.

### Details ...

- on SFCs are found in the *Instruction list*, for details refer to the *Online Help for STEP 7* or to the *System and Standard Functions* Reference Manual.
- on communication are found in the *Communication with SIMATIC* Manual.

## 3.6 S7 Connections

### Introduction

S7 module intercommunication requires an S7 connection representing the communication path. Global data communications and PtP communications do not require an S7 connection.

All active communications require CPU S7 connection resources. Every S7 CPU provides a specific number of S7 connection resources, used by diverse communication services (PG/OP communication, S7 communication or S7 Basic communication).

### End Point of an S7 Connection

An S7 connection always has two end points, namely one active and one passive end point:

- The active connection end point is assigned to the module establishing the S7 connection.
- The passive connection end point is assigned to the module accepting the S7 connection.

Every communication-capable module can be S7 connection end point. At the connection end point, an established communication always uses **one** S7 communication resource of the respective module.

### Transition point of an S7 Connection

An S7 connection between two communication-capable modules can be established across several subnets. Two subnets are interconnected via network node. The module operating as node is a Router, representing the transition point for an S7 connection.

Every CPU 31xC-2 DP can operate as S7 router. It can establish up to four routed connections without restricting the quantity frame of the S7 connection.

## Assigning S7 Connections

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

**Reservation in your configuration program**

- *STEP 7* automatically reserves one S7 connection per PG communication, respectively OP communication in a CPU inserted during hardware configuration.
- In *STEP 7* you can reserve S7 connections for PG / OP / S7 basic communication.

**Assigning Connections in the User Program**

S7 basic communication is established by the user program. The CPU's operating system initiates the connection and assigns the respective S7 connection.

**Assigning connections during commissioning, testing and diagnostics**

The online function of the engineering station (PG/PC with *STEP 7*) is used to assign S7 connections for PG communication:

- An S7 connection for PG communication you have reserved in your CPU hardware configuration is, therefore, assigned to this engineering station.
- If all reserved S7 communications for PG communication are occupied, the operating system assigns the free S7 connections which have not yet been reserved. If no free connection is available the engineering station cannot communicate online with the CPU.

**Assigning Connections to B&B Services**

An Online function of the B&B station (OP/TP/... with *ProTool*) uses S7 connections for OP communication:

- An S7 connection for OP communication you have reserved in your CPU hardware configuration is therefore assigned to the B&B station engineering station.
- If all reserved S7 communications for OP communication are occupied, the operating system assigns non-reserved and free S7 connections. If no free connection is available, the B&B station cannot communicate online with the CPU.

## Time sequence for allocating S7 connections

Parameter assignment blocks are generated during configuration in *STEP 7*. They are called up on startup of the module. Here the module's operating system reserves or assigns the respective S7 connections. This implies, for example, that an operator station cannot access a reserved S7 connection for PG communication.

The module's S7 connections which are not reserved (CPU) can be used freely. These S7 connections are assigned in the order they are called.

**Example:**

If the CPU has only one more S7 connection available, the user can connect a PG to the bus for communication with the CPU. The S7 connection is only occupied if PG communication is active.
An OP establishes communication with the CPU when the user connects it to the bus while PG communication is down. Since the OP communication is continuous, the PG cannot go online to the CPU anymore.

## Distribution of S7 Connections

The following table shows the distribution of S7 connections of 31xC CPUs:

Table 3-8 Distribution of S7 connections of CPUs 31xC

| Communication Service | Distribution |
|---|---|
| PG communication<br>OP communication<br>S7 basic communication | In order to make the allocation of connection resources dependent not only on the chronological sequence in which various communication services are registered, S7 connection resources can be reserved for the following services.<br>For PD/OP communication, at least one connection resource is reserved as the default setting. Lower values are not possible.<br>In the table below, and in the CPU's technical data, you can find the configurable S7 connections as well as the default configuration for the CPUs. "Redistribution" of S7 connections is specified in your *STEP 7* configuration of the CPU. |
| S7 communication<br>Other communication resources (e.g. via CP 343-1, with a data length of > 240 bytes) | Here you can assign free S7 connection resources which have not been reserved for a specific service (PG/OP communication, S7 Basic communication. |
| Routing of PG functions<br>(only CPU 31xC-2 DP) | The CPUs provide four connections for routing of PG functions, available in addition to S7 connections. |
| Global data communication<br>Point-to-point connection | These communication services do **not** use S7 connections. |

## S7 connections of CPUs 31xC

The following table shows S7 connections available on the specific CPUs.

Table 3-9 S7 connections of CPUs 31xC

| Parameters | CPU 312C | CPU 313C<br>CPU 313C-2 DP/PtP | CPU 314C-2 DP/PtP |
|---|---|---|---|
| Total number of S7 connections | 6 | 8 | 12 |
| • reserved for PG communication | 1 to 5<br>Default: 1 | 1 to 7<br>Default: 1 | 1 to 11<br>Default: 1 |
| • reserved for OP communication | 1 to 5<br>Default: 1 | 1 to 7<br>Default: 1 | 1 to 11<br>Default: 1 |
| • reserved for S7 basic communication | 0 to 2<br>Default: 2 | 0 to 4<br>Default: 4 | 0 to 8<br>Default: 8 |
| • free S7 connections | All S7 connections which are not reserved are shown as free connection. | | |

## Example of a CPU 314C-2 DP

The CPU 314C-2 DP provides 12 S7 connections:

- You reserve two S7 connections for PG communication.
- You reserve two S7 connections for OP communication.
- You reserve two S7 connections for S7 basic communication.

This leaves six S7 connections available for any communication service, e.g. S7 communication, OP communication etc.

## Details ...

- on SFCs are found in the *Instruction list*, for details refer to the *Online Help for STEP 7* or to the *System and Standard Functions* Reference Manual.
- on communication are found in the Manual *Communication with SIMATIC*.
- on routing are found in the chapter below and in the *Online Help for STEP 7*.

## 3.7 Routing

### PG/PC access to stations on other subnets

As of *STEP 7* V5.1 + SP 2 it is possible to access S7 stations of other subnets with the PG/PC, e.g. to load user programs or hardware configurations, or to execute testing and diagnostic functions. You can connect the PG at any available point in the network and establish a connection to all stations available via network node.

The CPUs provide four connections for routing of PG functions, available in addition to S7 connections.

**Routing is not possible with a CPU 31xC that has been configured for operation as intelligent slave!**

### Network transition

Transitions between subnets are routed in a SIMATIC station that is equipped with interfaces to the respective subnets.

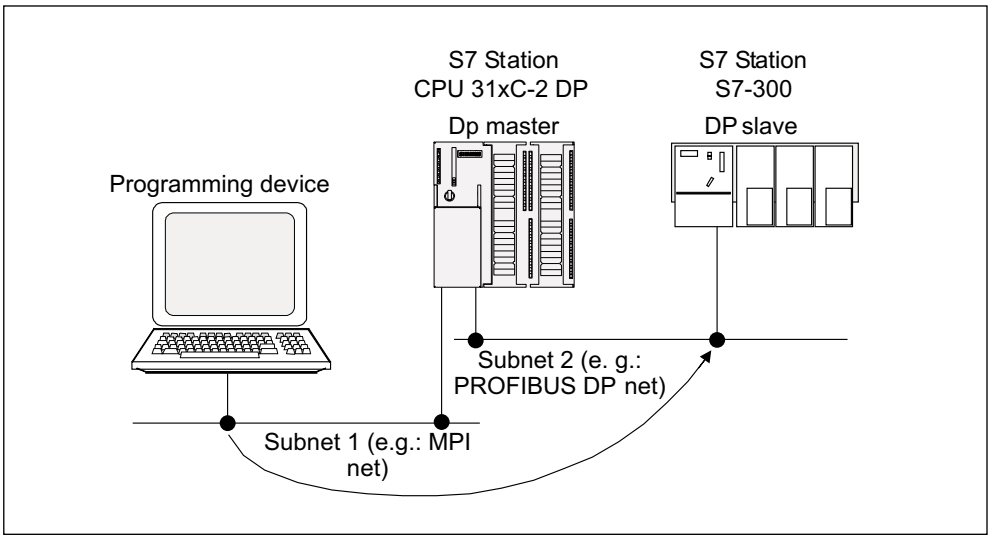



Figure 3-4 Routing - Network transition

### Prerequisites

- The station modules must be capable of routing (CPUs or CPs).
- The CPU 31xC must be configured as master.
- The network configuration does not exceed project limits.
- Configuration data containing the complete "knowledge" of the project's network configuration must have been downloaded to the modules.

  Reason: All internetworking modules must contain data on available subnets and network paths (= routing information).
- In your network configuration, the PG/PC you want to use to establish a routed communication must be assigned to the network it is physically connected to.

### Sample application: TeleService

The following sample application demonstrates the use of a PG for remote maintenance of an S7 station. Here, a modem is used to establish communication across subnet boundaries.

The lower part of the figure shows you how easy it is to configure this feature in *STEP 7*.

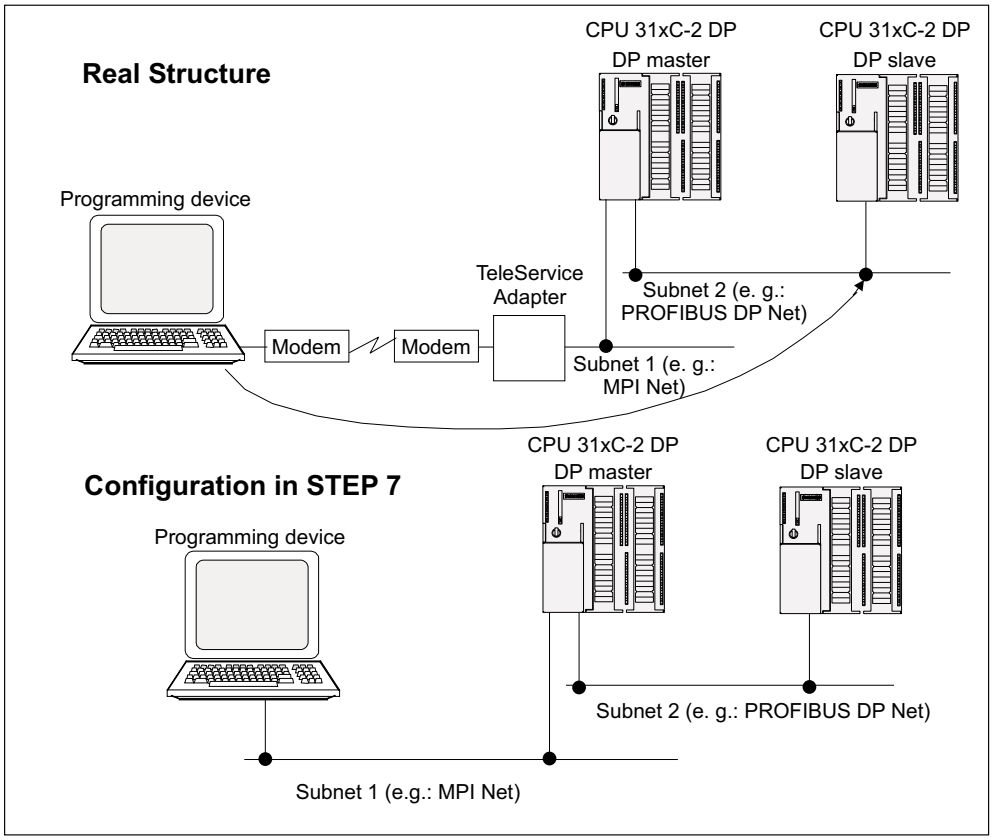



Figure 3-5 Routing - Sample application TeleService

### Further information ...

- on configuration with *STEP 7* is found in the Manual *Configuring Hardware and Connections with STEP 7*.
- of basic character is found in the Manual *Communication with SIMATIC*.

## 3.8 Data Consistency

A data area is consistent, if the operating system can read/write access data in a single block. Data exchanged collectively between the stations should belong together and originate from a single processing cycle, that is, be consistent.

If there is a programmed communication function such as XSEND/ XRCV which accesses shared data, then access to that data area can be coordinated by means of the parameter "BUSY“ itself.

### With PUT/GET functions

However, in the case of S7 communication functions such as PUT/GET or Read/Write using OP communication which do not require a block in the user program of the CPU 31x (as server), the extent of data consistency must be taken into account during the actual programming.

The PUT/GET functions for S7 communication or, as the case may be, the reading/writing of variables using OP communication are processed at the scan cycle checkpoint of the CPU.

To ensure a defined process interrupt response time, communication variables are copied consistently to/out of user memory in blocks of a maximum size of 32 bytes during the scan cycle checkpoint of the operating system. Data consistency is not guaranteed for data areas that are any larger.

Therefore, if a defined level of data consistency is required, the length of communication variables in the user program must exceed 32 bytes.

# Memory Conception

# 4

## 4.1 Memory Areas of CPUs 31xC

### 4.1.1 Distribution

#### Introduction

CPU 31xC memory can be split into three areas:

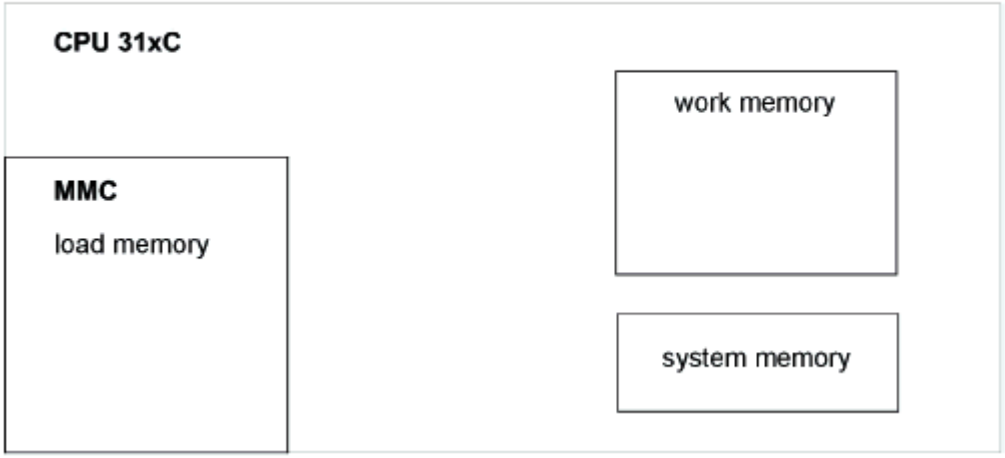



Figure 4-1 Memory Areas of a CPU 31xC

### Load memory

The load memory is located on a SIMATIC Micro Memory Card (MMC). Its size corresponds exactly with that of the MMC. It is used for storing code blocks and data blocks, as well as system data (configuration, connections, module parameters etc.).

Blocks marked as being runtime irrelevant are stored exclusively in load memory.

You can also store the entire configuration data of your project on MMC.

---

**Note**

Loading the user program and, therefore, operation of a CPUs 31xC is **only possible with an inserted MMC**.

---

### Main memory

The integrated CPU work memory is not expandable. Its only purpose is processing of code and user program data. Programs are processed only in main memory and system memory.

CPU main memory is stored retentive when the MMC is inserted.

### System memory

The integrated CPU system memory is not expandable.

It contains

- the address areas for memory bits, timers and counters
- the I/O process image
- local data

## 4.1.2 Retentivity

### Introduction

Your CPU 31xC memory is retentive . Retentivity is realized on MMC and in the CPU.

Data is retained in retentive memory even after POWER OFF and on restart (warm start).

### Load memory

Program data in load memory (MMC) is always retentive. It is written to MMC when loaded and it is protected against power loss and memory reset.

### Work memory

On POWER OFF your data is saved to MMC memory. Therefore, the content of data blocks is basically retentive.

### System memory

In your configuration (CPU properties, retentivity tab) you specify which part of memory bits, times and counters should be retentive and initialized with "0" on restart (warm restart).

Generally, the diagnostic buffer, MPI address (and transmission rate) and operating hour counter are written to retentive CPU memory. Retentivity of the MPI address and transmission rate ensures that your CPU is still capable of communication even after power loss, memory reset or loss of communication parameters (e.g. removal of the MMC or deletion of communication parameters).

### Retentive Behavior of Memory Objects

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

Table 4-1 Retentive behavior of memory objects

| Memory Object | Operating state transition | | |
|---|---|---|---|
| | **POWER ON / POWER OFF** | **STOP → RUN** | **Memory Reset** |
| User program/data (Load memory) | x | x | x |
| Actual value of the DBs | x | x | – |
| memory bits, times and counters configured as being retentive | x | x | – |
| diagnostics buffer, operating hour counter | x | x | X |
| MPI address, transmission rate | x | x | X |

x = retentive; – = not retentive

## 4.2 Memory functions

### Introduction

Memory functions help you to generate, edit or delete user programs or specific function blocks. You should also use the option of archiving your project data, in order to ensure data retentivity.

### General: Downloading the user program from PG/ PC

The PG/PC is used to download the complete user program **via MMC**. This operation might delete all function blocks stored in load memory.

Function block areas are specified under "Load memory requirements" in "General block properties".

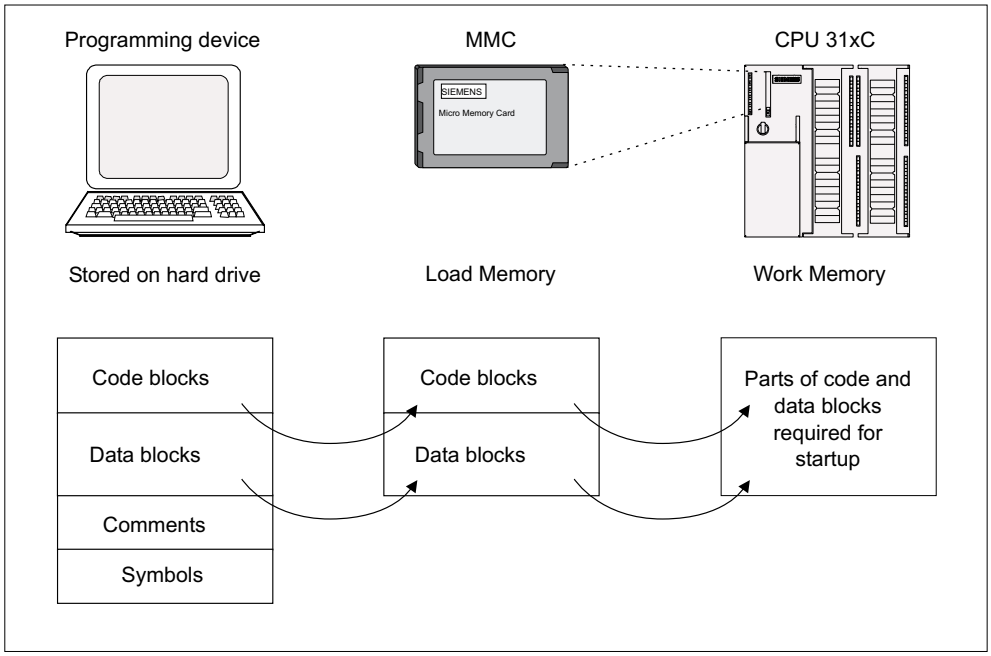



Figure 4-2 Load/Work memory

The program can only be started after all function blocks have been loaded.

**Note**

This function is only permitted in CPU STOP mode.

Load memory is cleared if this load operation was interrupted as a result of power loss or illegal function blocks.

## Downloading a user program from PG/ PC to MMC

**Case A: Loading a new user program**

You have created a new user program. Using a PG/ PC, download the complete program to MMC.

**Case B: Adding function blocks**

You have created a user program and downloaded it to MMC (Case A). You then want to add function blocks to the program. In this case you do not need to reload the complete user program to MMC. Rather, you only download the new function blocks to MMC (this procedure reduces the time required for loading highly complex programs!).

**Case C: Overloading**

In this case you edit the function blocks of your user program. In the next step, overload the user program or only the changed function blocks to the MMC, using the PG/PC.

**Warning**

When overloading functions blocks/user programs, all data stored under the same name is lost on the MMC.

After a block is loaded, data of runtime relevant blocks is transferred to memory and activated.

## Deleting blocks

Blocks are deleted in load memory. In *STEP 7* They can be deleted by a user program instruction (DBs also with SFC 23 "DEL_DB").

Memory area used by this block is released.

## Uploading

In contrast to load operations, upload is defined as uploading a specific block or a complete user program **from the CPU to the PG/PC**. In this case the block content represents data previously loaded to the MMC. Here, runtime relevant DBs form the exception; their actual values are transferred.

In *STEP 7*, CPU memory allocation is not influenced by an upload of blocks or of the user from the CPU.

## Compression

Compression is used to reorganize memory space which has been fragmented as a result of load/delete operations, thus reorganizing free memory area in a continuous block.

Compression is possible in CPU STOP or RUN mode.

## Writing to RAM to ROM (RAM to ROM)

In this write RAM to ROM operation, the actual values of the DBs in main memory are written to load memory as initial values.

---

**Note**

This function is only permitted in CPU STOP mode.

There will subsequently be no data in load memory if this operation is interrupted by power failure.

---

## Removing/Inserting the MMC

The CPU cannot operate in RUN mode if the MMC is missing (no load memory). Appropriate operation is not possible until after an MMC is inserted and memory is reset.

The CPU recognizes removal and insertion of an MMC in any operating state.

**Removal procedure:**

1. The CPU must be switched to STOP mode.
2. All writing PG functions must be disabled (e.g. loading of blocks)
3. After you have inserted the MMC, the CPU prompts you to perform a memory reset.

---

**Warning**

Data on a SIMATIC Micro Memory Card can be corrupted if you remove the card during write access. In this case and if required, MMC memory must be deleted via PG or formatted in the CPU.

Never remove an MMC in RUN mode. Always remove it in power off or CPU STOP state, when the PG does not write access the card. Disconnect the communication lines if you cannot safely exclude active write access functions from the PG (e.g. load/delete function block).

---

**Insertion sequence:**

Insert an MMC with the corresponding user program as follows:

1. Insert the MMC
2. The CPU requests a memory reset
3. Acknowledge memory reset

   The CPU might request another memory reset if you have inserted the wrong MMC or an MMC with firmware update. In this case, refer to Chapter *Structure and Communication Connections of CPU 31xC, Special Handling* for a description of corresponding procedures.
4. Starting the CPU

**Warning**

Make sure that the MMC contains an application program matching your CPU (system). A wrong user program can cause fatal processes.

## Memory Reset

After insertion/removal of a Micro Memory Card, memory reset establishes defined conditions to enable a CPU restart (warm start). Memory reset rearranges the CPU's memory management. All function blocks in load memory are retained. All runtime relevant function blocks are copied once again from load memory to main memory and, in particular, the data blocks in main memory are initialized (reset to initial values). Memory reset and the corresponding peculiarities are described in the S7-300 Installation Manual, Chapter *CPU Memory Reset*.

## Restart (warm start)

- The actual values of all DB are retained.
- The value of all retentive M, C, T is maintained.
- All non-retentive application data is initialized:
  - M, C, T, I, O with "0"
- All runtime levels are initialized.
- The process images are deleted.

## 4.3 Address areas

### Overview

S7 CPU system memory is split into address areas (refer to the table below). In a corresponding operation of your user program, you address data directly in the respective address area.

Table 4-2 Address Areas of System Memory

| Address areas | Description |
|---|---|
| Input Process Image | At the start of every OB 1 cycle, the CPU copies the input values of the input modules to the input process image. |
| Output Process Image | During a program cycle the output values are calculated and written to the output process image. At the end of the OB 1 cycle, the CPU writes those calculated values to the output modules. |
| Bit memories | This area provides memory for intermediate results of a program calculation. |
| Timers | Timers are available in this area. |
| Counter | Counters are available in this area. |
| Local data | During code block (OB, FB, FC) processing, temporary data of the corresponding block is saved to this memory area. |
| Data blocks | Refer to Chapter *Handling of Data in a DB* |

Address areas possible for your CPU are listed in *S7-300 Instruction list for CPUs 31xC*.

### I/O Process Image

When addressing Input (I) and Output (O) address areas, the user program does not query the signal state of digital signal modules, it rather accesses a memory area in CPU system memory. This memory area is defined as process image.

This process image is split into two sections: Input and Output.

**Advantages of the Process Image**

Process image access, compared to direct I/O access, offers the advantage that a consistent image of process signals is made available to the CPU during cyclic program processing. 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 at the next cycle. Moreover, since the process image is stored in CPU system memory, access is significantly faster than direct access to signal modules.

### Process Image Update

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

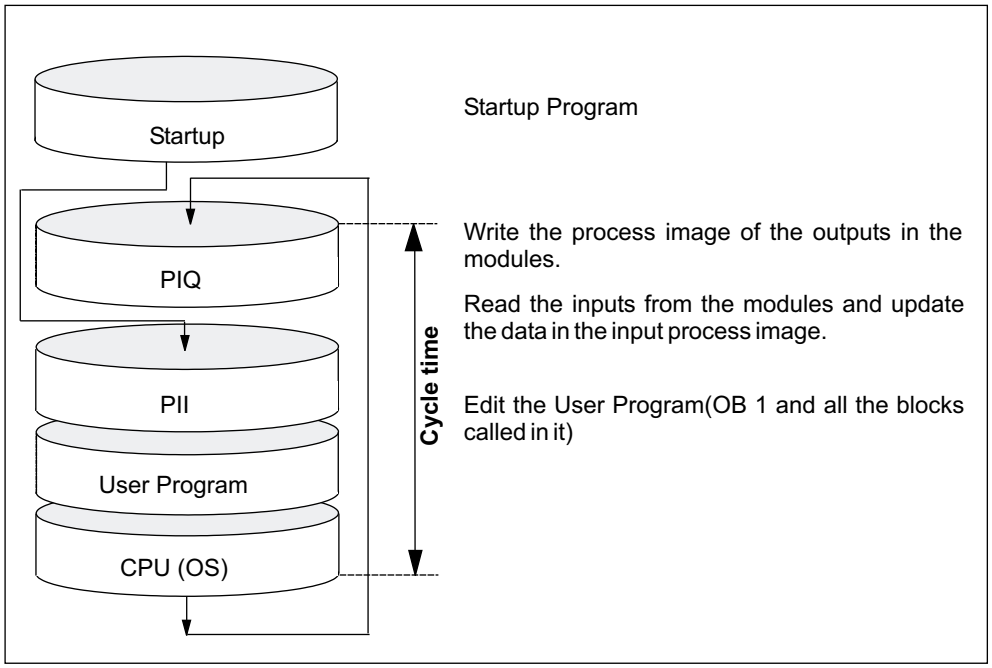



Figure 4-3 Sequence of operation within a cycle

## Local Data

Local data contains:

- Temporary code block variables
- OB start information
- Transfer parameters
- Intermediate results

### Temporary Variables

When you create blocks, you can declare temporary variables (TEMP) which are only available during block processing and subsequently overwritten. Local data is of a fixed length in each OB. They must be initialized prior to an initial read access. The OB also requires 20 bytes of local data for its start information. Local data access is faster than access to the data in DBs.

The CPU is equipped with memory for storing the temporary variables (local data) of currently processed blocks. The size of this memory area depends on the CPU. It is allocated in partitions of equal size to the priority classes. Every priority class has its own local data area.

**Caution**

All temporary variables (TEMP) of an OB and its subordinate blocks are stored in local data. The use of multiple nesting levels for block processing can cause overflow of local data memory.
The CPUs will change to STOP mode if you exceed the permissible size of local data for a priority class.
Take the local data requirement of synchronous error OBs into account; it is allocated to the respective triggering priority class.

## 4.4 Handling of DB Data

### 4.4.1 Recipes

#### Introduction

A recipe represents a collection of user data.

You can realize a simple recipe conception using DBs which are runtime irrelevant. In this case the recipes should be of the same structure (length). One DB should exist per recipe.

#### Processing sequence

**The recipe is to be stored in load memory:**

- The specific data records are generated in *STEP 7* as runtime irrelevant DB and then downloaded to the CPU. Therefore, recipes utilize load memory, rather than main memory.

**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 runtime irrelevant DB in main memory. This operation reduces main memory load to the data quantity of one data record.

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

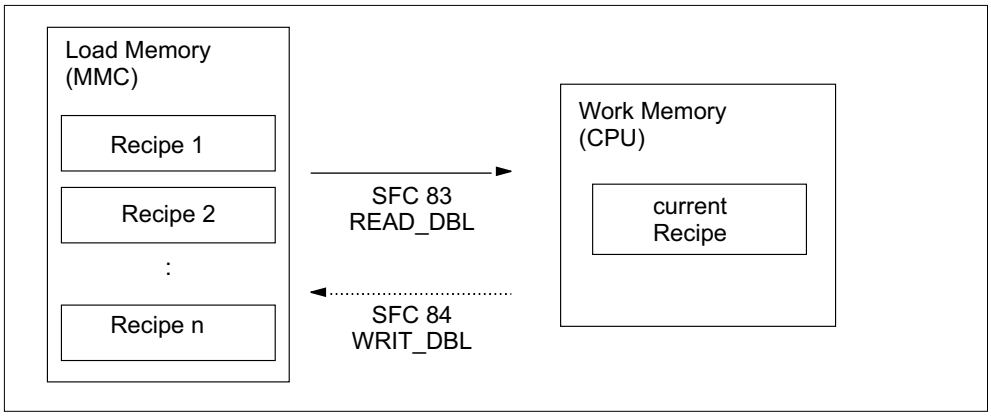



Figure 4-4 Handling of Recipe Data

**Writing back a changed recipe:**

- SFC 84 "WRIT_DBL" can be called in the user program to write back new or changed recipe data records generated during program processing to load memory.

  This data written to load memory are portable and also retentive on memory reset.

  To create a backup, upload modified data records (recipes) in a single block to the PG/PC.

---

**Note**

Active system functions of SFC 82 to 84 (current accesses to the MMC) have a distinct influence on PG functions (e.g. Block Status, Variable Status, Load Block, Upload, Open).

Here, performance is typically reduced (compared to inactive system functions) by the factor 10.

---

**As a precaution against data loss, always make sure that the maximum number of delete/write operations is not exceeded. Also refer to the Chapter "Structure and Communication Connection of CPUs 31xC", SIMATIC Micro Memory Card (MMC).**

---

**Caution**

Data on a SIMATIC Micro Memory Card can be corrupted if you remove the card during write access. In this case and if required, MMC memory must be deleted via PG or formatted in the CPU. Never remove an MMC in RUN mode. Always remove it in power off or CPU STOP state, when the PG does not write access the card. Disconnect the communication lines if you cannot safely exclude active write access functions from the PG (e.g. load/delete function block).

---

## 4.4.2 Measurement Value Archive

### Introduction

Measurement values are generated when the CPU processes the user program. These values are to be evaluated and archived.

### Processing sequence

**Accumulating Measurement Values:**

- The CPU accumulates measured values in one DB (for alternating backup mode in several DBs) located in main memory.

**Archiving Measurement Values:**

- You can call SFC 84 "WRIT_DBL" in the user program to swap measured values stored in the DB to load memory, before the data volume can exceed main memory capacity.

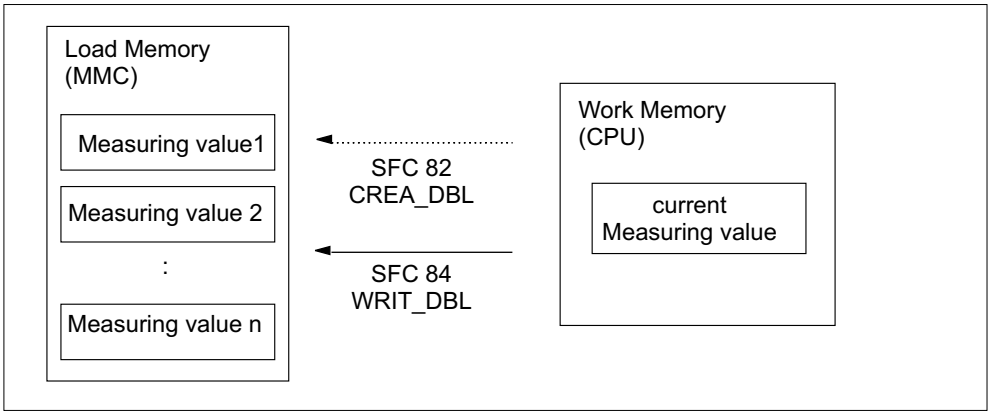



Figure 4-5 Handling of Measurement Value Archives

- You can call SFC 82 "CREA_DBL" in the user program to generate new (additional) run-time irrelevant DBs in load memory that do not require main memory.

---

**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 main memory.

---

This data written to load memory are portable and also retentive on memory reset.

**Evaluation of Measurement Values:**

- Measurement value DBs can be uploaded from load memory to other communication partners (e.g. PG, PC, ...), where they can be evaluated.

---

**Note**

Active system functions of SFC 82 to 84 (current accesses to the MMC) have a distinct influence on PG functions (e.g. Block Status, Variable Status, Load Block, Upload, Open).

Here, performance is typically reduced (compared to inactive system functions) by the factor 10.

---

**As a precaution against data loss, always make sure that the maximum number of delete/write operations is not exceeded. Also refer to the Chapter "Structure and Communication Functions of CPUs 31xC", SIMATIC Micro Memory Card (MMC).**

---

**Caution**

Data on a SIMATIC Micro Memory Card can be corrupted if you remove the card during write access. In this case and if required, MMC memory must be deleted via PG or formatted in the CPU.

Never remove an MMC in RUN mode. Always remove it in power off or CPU STOP state, when the PG does not write access the card. Disconnect the communication lines if you cannot safely exclude active write access functions from the PG (e.g. load/delete function block).

---

## 4.5 Description of SFC 82 to SFC 84

### 4.5.1 Creating a DB in load memory with SFC 82 "CREA_DBL"

#### Description

With SFC 82 "CREA_DBL" you create data a new data block in the MMC load memory. SFC 82 generates a DB with a number from a specified area and of default size. SFC 82 assigns the lowest possible number of this area to the DB. You can generate a specific number by entering the same value for the upper and lower area limit. You cannot assign numbers already assigned to the DBs in your user program. The SFC is terminated and an error message is generated if a DB with identical number already exists in main/load memory or in a copied version.

**Note**

You can use SFC 24 "TEST_DB" to determine whether an identical DB number already exists.

Parameter SRCBLK points to the source area that contains the data to be written to the DB. This data area must be a DB or DB area. This data area must not be changed when processing SFC 82 in order to maintain data consistency.

A DB with READ_ONLY attribute can only be created and initialized by SFC 82.

SFC 82 does not change the checksum of your user program.

## Description

SFC 82 "CREA_DBL" operates asynchronously, that is, processing covers several SFC calls. Start the request by calling SFC 82 with REQ = 1. The job status is displayed in output parameter RET_VAL and BUSY. Also refer to the description of REQ, RET_VAL and BUSY with asynchronously operating SFCs, in the Reference Manual *System and Standard Functions*.

| Parameter | Declaration | Data Type | Memory area | Description |
|---|---|---|---|---|
| REQ | INPUT | BOOL | I, O, M, D, L | REQ = 1: request to generate a DB |
| LOW_LIMIT | INPUT | WORD | I, O, M, D, L | Lower limit of the area from which the SFC fetches your DB number |
| UP_LIMIT | INPUT | WORD | I, O, M, D, L | Upper limit of the area from which the SFC fetches your DB number |
| COUNT | INPUT | WORD | I, O, M, D, L | The count value specifies the number of data bytes you want to reserve for your DB. Here you must specify an even number of bytes. |
| ATTRIB | INPUT | Byte | I, O, M, D, L | DB properties:<br>Bit 0 = 1: UNLINKED:<br>DB is only in load memory.<br>Bit 1 = 1: READ_ONLY:<br>DB is write protected.<br>Bit 2 = 1: NON_RETAIN:<br>DB is not retentive. *)<br>Bit 3 to 7: reserved |
| SRCBLK | INPUT | ANY | D | Pointer to the data area used to initialize the DB |
| RET_VAL | OUTPUT | INT | I, O, M, D, L | Error code |
| BUSY | OUTPUT | BOOL | I, O, M, D, L | BUSY = 1: process not yet closed. |
| DB_NUM | OUTPUT | WORD | I, O, M, D, L | Number of the generated DB |

*) The NON_RETAIN attribute is not available at present.

## Error information

| Error ID (W#16#...) | Description |
|---|---|
| 0000 | No error |
| 0081 | Target area is larger than source area.<br>All source data is written to the target area. The remaining bytes are filled with "0". |
| 7000 | Initial call with REQ = 0: no data transfer active; BUSY value is "0". |
| 7001 | Initial call with REQ = 1: data transfer is initiated; BUSY value is "1". |
| 7002 | Intermediate call (REQ irrelevant): data transfer already active; BUSY value is "1". |
| 8081 | Source area is larger than target area.<br>Target area is written completely, the remaining source bytes are ignored. |
| 8091 | You have performed a nested call of SFC 82. |
| 8092 | The "Create DB" function cannot be carried out at present, because<br>• the CPU does not currently provide the required resources<br>• the "Compress user memory" function is currently active<br>• the H-CPU is currently establishing a connection or executing an update |
| 8093 | Parameter SRCBLK (Initialization block) does not specify a runtime relevant DB. |
| 8094 | Parameter ATTRIB specifies an attribute currently not supported . |
| 80A1 | DB number error:<br>• The number is "0".<br>• Lower limit > Upper limit |
| 80A2 | DB length error:<br>• The length is "0".<br>• The length is an odd number.<br>• The length is greater than permitted by the CPU. |
| 80B1 | No DB available. |
| 80B2 | Out of main memory |
| 80B3 | Out of continuous memory (perform a compression) |
| 80BB | Out of load memory |
| 80C0 | The target is currently being processed by another SFC or communication function. |
| 80C3 | The required operation resources are currently in use. |
| 8xyy | General error codes, e.g.:<br>• The source DB does not exist or is only available in a copied version<br>• Source area does not exist in the DB |

x is the respective parameter number.

## 4.5.2 Reading from a DB in load memory using SFC 83 "READ_DBL"

### Description

SFC 83 "READ_DBL" (read data block in load memory) is used to read a DB or a DB area from load memory (SIMATIC Micro Memory Card), and write the data to the data area of the target DB.

The target DB must be runtime relevant (keyword UNLINKED = 0). The source area to be read can also be runtime irrelevant (keyword UNLINKED = 1). The content of load memory is not changed by the read operation.

This data area must not be changed when processing SFC 83 in order to maintain data consistency.

The following restrictions apply to the parameters SRCBLK and DSTBLK:

- The length value of an ANY pointer of the type BOOL must be divisible by 8.
- The length of an ANY pointer of the type STRING must be equal to 1.

If required, you can determine the length of the source DB by calling SFC24 "TEST_DB".

---

**Note**

SFC 83 is processed asynchronously. Therefore, it is not suitable for frequent read access to variables in load memory.

Once a job has started it is always completed, even if this resource is requested with higher priority. If error code 80C3 is displayed in a higher-priority run level it would therefore make no sense to restart the request immediately. You should rather wait until the blocking job has terminated itself.

---

## Description

SFC 83 "CREA_DBL" operates asynchronously, that is, processing covers several SFC calls. Start the request by calling SFC 83 with REQ = 1. The job status is displayed in output parameter RET_VAL and BUSY. Also refer to the description of REQ, RET_VAL and BUSY with asynchronously operating SFCs, in the Reference Manual *System and Standard Functions*.

| Parameter | Declaration | Data Type | Memory area | Description |
|---|---|---|---|---|
| REQ | INPUT | BOOL | I, O, M, D, L | REQ = 1: request to read |
| SRCBLK | INPUT | ANY | D | Pointer to the data area to be read of the DB in load memory |
| RET_VAL | OUTPUT | INT | I, O, M, D, L | Error code |
| BUSY | OUTPUT | BOOL | I, O, M, D, L | BUSY = 1: read process not yet terminated. |
| DSTBLK | OUTPUT | ANY | D | Pointer to the data area of the target DB |

## Error information

| Error ID (W#16#...) | Description |
|---|---|
| 0000 | No error |
| 0081 | Target area is larger than source area.<br>The complete source area is written to the target area, the remaining bytes of the target area are not changed. |
| 7000 | Initial call with REQ = 0: no data transfer active; BUSY value is "0". |
| 7001 | Initial call with REQ = 1: data transfer is initiated; BUSY value is "1". |
| 7002 | Intermediate call (REQ irrelevant): data transfer already active; BUSY value is "1". |
| 8081 | Source area is larger than the target area.<br>Target area is written completely, the remaining source bytes are ignored. |
| 8092 | The "Read DB" function cannot be carried out at present, because<br>• The CPU does not currently provide the required resources<br>• The "Compress user memory" function is currently active<br>• The H-CPU is currently establishing a connection or executing an update |
| 8093 | Parameter DSTBLK does not specify a runtime relevant DB. |
| 80B4 | DB with F attribute must not be changed. |
| 80C0 | The target is currently being processed by another SFC or communication function. |
| 80C3 | The required operating resources are currently in use. |
| 8xyy | General error codes, e.g.:<br>• The source DB does not exist or is only available in a copied version<br>• Source area does not exist in the DB |

x is the respective parameter number.

### 4.5.3 Writing to a data block in load memory, using SFC 84 "WRIT_DBL"

#### Description

Use SFC 84 "WRIT_DBL" (write data block in load memory) to write the content of a source DB to a DB or to the area of a DB on load memory (SIMATIC Micro Memory Card).

The DB in load memory parameter DSTBLK refers to can be runtime relevant or not. The source area parameter SRCBLK refers to can be a DB or a DB(content) in main memory. Therefore, the source DB parameter SRCBLK refers to must be runtime relevant (keyword UNLINKED = 0). The source DB can also be generated with SFC 22 "CREAT_DB".

In order to maintain data consistency, this source area must not be changed when processing SFC 84 , .

The following restrictions apply to the parameters SRCBLK and DSTBLK:

- The length value of an ANY pointer of the type BOOL must be divisible by 8.
- The length of an ANY pointer of the type STRING must be equal to 1.

If required, you can determine the length of the target DB by calling SFC24 "TEST_DB".

SFC 84 does not change the version ID of the user program if you write to a DB created by means of an SFC. Initial write access to a loaded DB changes the checksum of the user program.

---

**Note**

SFC 84 is processed asynchronously. Therefore, it is not suitable for frequent write access to variables in load memory. Moreover, frequent write access reduces the service life of the MMC.

---

## Description

SFC 84 "WRIT_DBL" operates asynchronously, that is, processing covers several SFC calls. Start the request by calling SFC 84 with REQ = 1.

The job status is displayed in output parameter RET_VAL and BUSY.

Also refer to the description of REQ, RET_VAL and BUSY with asynchronously operating SFCs, in the Reference Manual *System and Standard Functions*.

| Parameter | Declaration | Data Type | Memory area | Description |
|---|---|---|---|---|
| REQ | INPUT | BOOL | I, O, M, D, L | REQ = 1:request to write |
| SRCBLK | INPUT | ANY | D | Pointer to the data area of the source DB |
| RET_VAL | OUTPUT | INT | I, O, M, D, L | Error code |
| BUSY | OUTPUT | BOOL | I, O, M, D, L | BUSY = 1: write process still busy. |
| DSTBLK | OUTPUT | ANY | D | Pointer to the data area to be written of the DB in load memory |

## Error information

| Error ID (W#16#...) | Description |
|---|---|
| 0000 | No error |
| 0081 | Target area is larger than source area.<br>The complete source area is written to the target area, the remaining bytes of the target area are not changed. |
| 7000 | Initial call with REQ = 0: no data transfer active; BUSY value is "0". |
| 7001 | Initial call with REQ = 1: data transfer is initiated; BUSY value is "1". |
| 7002 | Intermediate call (REQ irrelevant): data transfer already active; BUSY value is "1". |
| 8081 | Source area is larger than target area.<br>Target area is written completely, the remaining source bytes are ignored. |
| 8092 | The "Write DB" function cannot be carried out at present, because<br>• The CPU does not currently provide the required resources<br>• The "Compress user memory" function is currently active<br>• The H-CPU is currently establishing a connection or executing an update |
| 8093 | Parameter DSTBLK does not specify a runtime relevant DB. |
| 80B4 | DB with F attribute must not be changed. |
| 80C0 | The target is currently being processed by another SFC or communication function. |
| 80C3 | The required operating resources are currently in use. |
| 8xyy | General error codes, e.g.:<br>• The source DB does not exist or is only available in a copied version<br>• Source area does not exist in the DB |

x is the respective parameter number.

## 4.6 Saving/retrieving complete projects to/from Micro Memory Card

### How the functions operate

Using the **Save project to Memory Card** and **Fetch project from Memory Card**, you can save the complete project data to a SIMATIC Micro Memory Card for future retrieval. Here, the SIMATIC Micro Memory Card can be installed in a CPU or in the MMC programming device of a PG or PC.Functions for saving/retrieving project data to/from a SIMATIC Micro Memory Card are only available for CPUs 31xC.

---

**Note**

CPUs 31xC are currently not covered in the *STEP 31* Online Help. Thus, the functions **Save project to Memory Card** and **Retrieve project from Memory Card** are here only relevant for CPUs 41x.
CPUs 31xC provide the function scope described in this manual.

---

Project data is compressed before it is saved to a SIMATIC Micro Memory Card, and uncompressed on retrieval. The selected size of the Micro Memory Card must make allowances for saving project data as well as additional user data. The size of project data to be saved corresponds with the size of the project's archive file. A message informs you if the Micro Memory Card is out of memory.

For technical reasons only the complete content (user program and project data) can be transferred, if the target for **Save project to Memory Card** is a SIMATIC Micro Memory Card installed in a CPU.

## Handling the Functions

Handling of the **Save project to Memory Card** / **Retrieve project from Memory Card** functions depends on the location of the SIMATIC Micro Memory Card:

- In the project window of SIMATIC Manager, select a project level that is uniquely assigned to this CPU (e.g. CPU, program, source or blocks) if the Micro Memory Card is installed in the MMC slot of a CPU 31xC. Select the menu command **PLC > Save Project on the Memory Card** or **PLC> Get Project from Memory Card**. Now the complete project data is written to / retrieved from the Micro Memory Card.
- If project data is not available on the currently used programming device (PG/PC), you can select the source CPU in the window "Available nodes". Open the window "Available nodes" via menu item **PLC > Show available nodes** and select the connection/CPU that contains your project data on Micro Memory Card. Now select the menu item **Retrieve project from Memory Card**.
- If the Micro Memory Card is in the MMC prommer slot of a PG or PC, you can open the "S7-Memory Card window" via the menu command **File > S7-Memory Card > Open**. Select the menu command **PLC > Save Project on the Memory Card** or **PLC> Get Project from Memory Card**. to open a dialog in which you can select the source or target project.

**Project data can generate high data traffic. Especially in RUN mode with read/write access to the CPU, this can lead to waiting periods of several minutes.**

## Sample application

Once you have more than one member of service and maintenance staff occupied with a maintenance or service task on a SIMATIC PLC, it may be difficult to make current project data quickly available to each staff member. However, if these staff members have access to project data that is available locally on a serviced CPU, they can make their changes and quickly release the updated version to other staff members.

# Cycle and Response Times

# 5

## 5.1 Introduction

### In this chapter ...

we show you the structure of S7-300 cycle and response times.

You can read out he cycle time of your user program on the corresponding CPU with the programming device (see *Online Help on STEP 7* or manual *Configuring Hardware and Communications Connections STEP 7 V5.1*).

The samples below show you how to calculate the cycle time.

An important feature when looking at a process is the response time. In this chapter we will show you in detail how to calculate the response time.

### In this chapter ...

- Cycle Time
- Communication load
- Response Time
- Example of Calculating Cycle Time and Response Time
- Interrupt Response Time
- Calculation Example for the Interrupt Response Time
- Reproducibility of Delay/Watchdog Interrupts

### Further information on processing times ...

is found in the *S7-300 Instruction List for CPUs 31xC*, a chart containing all

- *STEP 7* instructions the respective CPU can process,
- in SFCs/SFBs integrated in the CPUs,
- in IEC functions which can be called in *STEP 7*.

## 5.2 Cycle Time

### 5.2.1 Overview

#### Introduction

This section explains the structure of the cycle time and how it is calculated.

#### Definition of Cycle Time

The cycle time represents the time the operating system requires for processing one program cycle, that is, one OB 1 cycle, including all program sections and system activities that interrupt this cycle.

This time is monitored.

#### Time sharing model

Cyclic program processing, and therefore user program processing, is executed in time shares. For better comprehension of the processes, we assume that every time share has a length of exactly 1 ms.

#### Process image

During cyclic program processing, the CPU requires a consistent image of the process signals. This is ensured by reading/writing the process signals prior to program processing. Subsequently, the CPU does not address input (I) and output (Q) address areas directly, but rather accesses system memory area that contains the I/O process image.

#### Sequence of cyclic program processing

The table and the figure below show the phases of cyclic program processing.

Table 5-1 Cyclic program processing

| Step | Sequence |
|---|---|
| 1 | The operating system starts cycle time monitoring. |
| 2 | The CPU writes the values of the output process image to the output modules. |
| 3 | The CPU reads the status at the input of the input modules and updates the input process image. |
| 4 | The CPU processes the user program in time intervals and executes the program's instructions. |
| 5 | At the end of the cycle the operating system executes tasks pending, e.g. loading and deleting blocks. |
| 6 | The CPU then returns to the cycle start and restarts cycle time monitoring. |

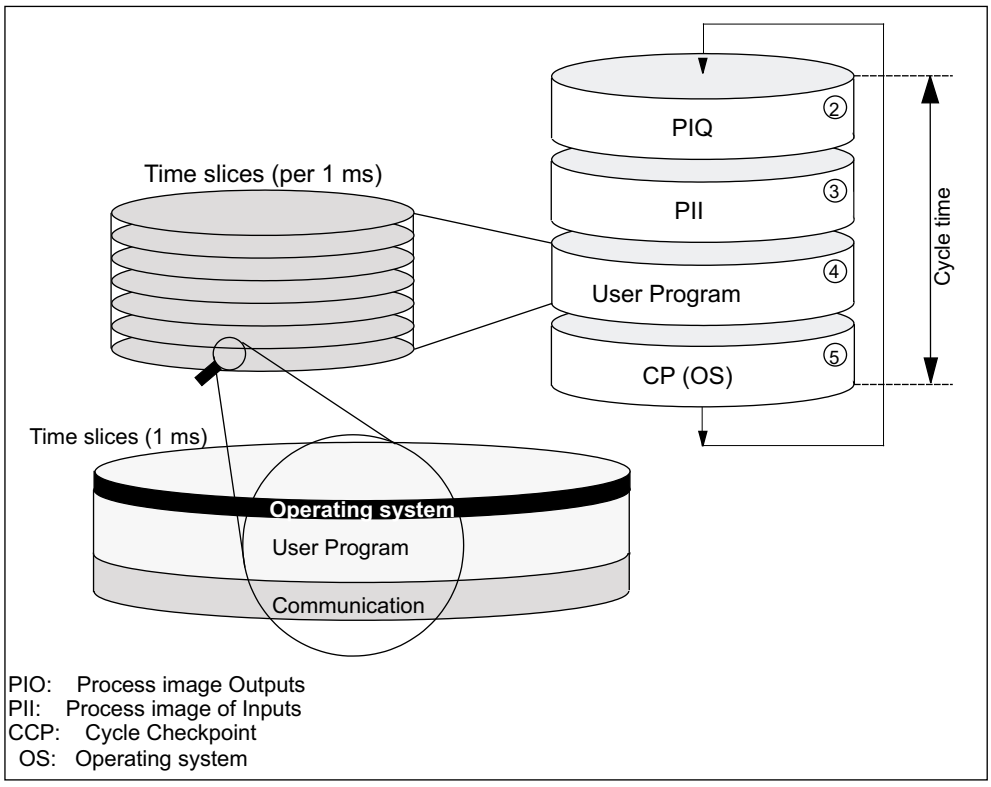



Figure 5-1 Time-sharing Model

## Extension of the Cycle Time

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

- Timecontrolled interrupt handling
- Process interrupt handling (also refer to Chapter *Interrupt response time*)
- Diagnostics and Error Handling
- Communication with programming devices (PGs), operator panels (OPs) and via connected CPs (e.g. Ethernet, PROFIBUS-DP)
- Testing and commissioning functions, e.g. status/controlling of variables or block status.
- Transferring and deleting blocks, compressing user program memory
- Writing/reading the MMC, using SFC 82 to 84 in the user program
- Technological functions
- PtP communication

## 5.2.2 Calculating the cycle time

### Introduction

The cycle time is the total of following influencing factors.

### Process image update

The table below shows the time a CPU requires to update the process image (process image transfer time). The times specified might be prolonged by interrupts or by communication of the CPU.

The transfer time for process image updates is calculated as follows:

BaseLoad (K) + number of bytes in the PI in the module 0 x (A)
+ number of bytes in the PI in the module 1 to 3 (B)
+ number of bytes in the PI via DP x (D)
**= transfer time for the process image**

Figure 5-2 Formula for calculating the process image (PI) transfer time

Table 5-2 Data for calculating the process image transfer time

| Constant | Components | CPU 312C | CPU 313C | CPU 313C-2 DP | CPU 313C-2 PtP | CPU 314C-2 DP | CPU 314C-2 PtP |
|---|---|---|---|---|---|---|---|
| K | Base load | 150 µs | 100 µs | 100 µs | | 100 µs | |
| A | per byte in module rack 0 | 37 µs | 35 µs | 37 µs | | 37 µs | |
| B | per byte in module rack 1 to 3 * | - | 43 µs | 47 µs | | 47 µs | |
| D (DP only) | per WORD in the DP area for the integrated DP interface | - | - | 1 µs | - | 1 µs | - |

* + 60 µs per module rack

## Extension of user program processing time

The table below lists the multiplication factors required for calculating your application program processing time.

Table 5-3 Extension of user program processing time

| CPU Sequence | 312C | 313C | 313C-2DP | 313C-PtP | 314C-2DP | 314C-2PtP |
|---|---|---|---|---|---|---|
| Factor | 1,06 | 1,10 | 1,10 | 1,06 | 1,10 | 1,09 |

## Operating system execution time at the scan cycle checkpoint

The table below shows the operating system execution time at the scan cycle checkpoint of the CPUs. These times apply without:

- Testing and commissioning functions, e.g. status/controlling of variables or block status
- Transferring and deleting blocks, compressing user program memory
- Communications
- Reading/writing the MMC with SFC 82 to 84

Table 5-4 Operating system execution time at the scan cycle checkpoint

| Sequence | CPU 312C | CPU 313C | CPU 313C-2 | CPU 314C-2 |
|---|---|---|---|---|
| Cycle control at the SCC | 500 µs | 500 µs | 500 µs | 500 µs |

## Cycle time extension as a result of nested interrupts and of errors

Activated interrupts also prolong the cycle time. Details are found in the table below.

Table 5-5 Extending the Cycle by Nesting Interrupts

| Interrupt type | CPU 312C | CPU 313C | CPU 313C-2 | CPU 314C-2 |
|---|---|---|---|---|
| Process interrupt | 700 µs | 500 µs | 500 µs | 500 µs |
| Diagnostic interrupt | 700 µs | 600 µs | 600 µs | 600 µs |
| Time of day interrupt | 600 µs | 400 µs | 400 µs | 400 µs |
| Delay Interrupt | 400 µs | 300 µs | 300 µs | 300 µs |
| Watchdog Interrupt | 250 µs | 150 µs | 150 µs | 150 µs |

The program run-time at interrupt level must be added to this extension.

Table 5-6 Cycle extension as a result of

| Type of error | CPU 312C | CPU 313C | CPU 313C-2 | CPU 314C-2 |
|---|---|---|---|---|
| Programming error | 600 µs | 400 µs | 400 µs | 400 µs |
| I/O access error | 600 µs | 400 µs | 400 µs | 400 µs |

The program runtime of the interrupt OB must be added to this extension.The corresponding times for multiple nested interrupt/error OBs are added.

**See also:** *How To Calculate Cycle/ Response Time*

## 5.2.3 Differing Cycle Times

### Overview

The cycle time ($T_{cyc}$) is not the same in every cycle. The figure below shows different cycle times $T_{cyc1}$ and $T_{cyc2}$ . $T_{cyc2}$ is longer than $T_{cyc1}$, because the cyclically processed OB 1 is interrupted by a time-of-day interrupt OB (here: OB 10).

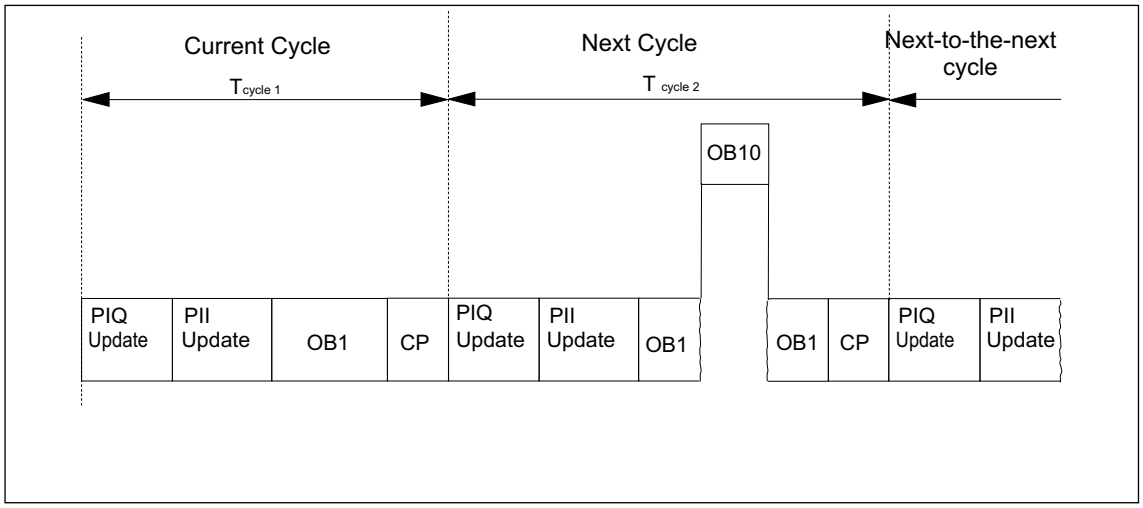



Figure 5-3 Differing Cycle Times

### Block processing time can fluctuate

Another factor influencing the length of cycle times is fluctuation of the block processing time (e.g. OB 1), due to:

- conditional instructions,
- conditional block calls,
- different program paths,
- loops etc.

### Maximum Cycle Time

In *STEP 7* you can change the default setting for the maximum cycle time. You can specify how the CPU responds to time errors in OB 80 that is called on expiration of this time.

The CPU switches to STOP mode if OB80 does not exist in its memory.

## 5.2.4 Communication load

### Configured communication load (PG/OP communication)

For communication, the CPU operating system continuously provides the configured percentage of total CPU processing performance (Time-sharing technology). Processing performance not required for communication is made available to other processes.

In HW Config, you can specify a communication load value between 5% and 50%. Default value is 20%.

You can use the following formula for calculating the cycle time extension factor:

$$\frac{100}{100 - \text{“projected Communication load in \%”}}$$

Figure 5-4 Formula for calculating communication load

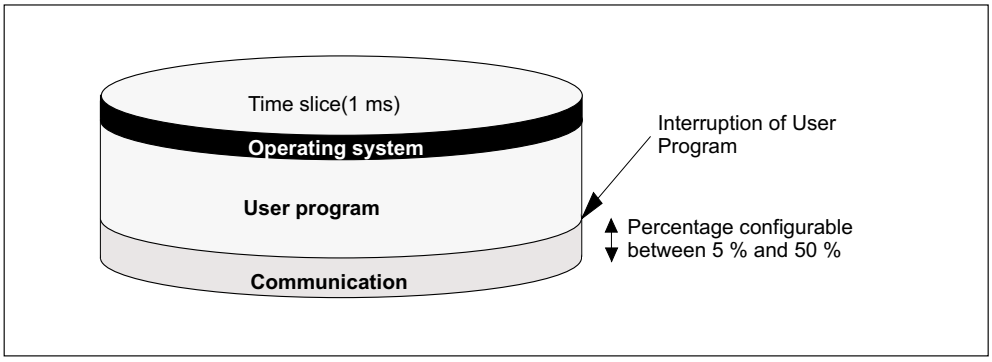



Figure 5-5 Splitting a time share

### 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.

### Real cycle time in dependence on communication load

The figure below describes the non-linear dependence of real cycle time on communication load. In our sample we have chosen a cycle time of 10 ms.

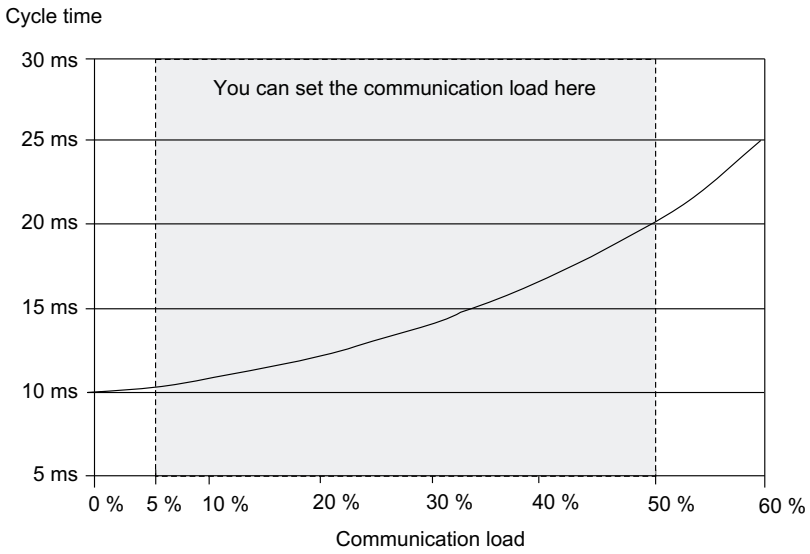



Figure 5-6 Dependence of the cycle time on communication load

### Influence on the physical cycle time

Statistically seen, during the OB1 cycle the occurrence of asynchronous events is more frequent than, for example, the number of interrupt events, due to the extension of cycle time by communication load. This causes an additional extension of the OB 1 cycle. This extension depends on the number of events occurring per OB 1 cycle and on the time required for processing these events.

**Note**

Check how a change of the "cycle load by communication parameter" influences system operation. Communication load must be taken into account when customizing the maximum cycle time. Otherwise, timeout errors can occur.

### Tips

- Use the default setting if possible.
- Increase this value only if the CPU is actually used for communications and if the user program is time critical.
- In all other cases you should only reduce this value.

## 5.2.5 Extension as a result of testing and commissioning functions

### Runtimes

The runtimes of testing and commissioning functions are operating system runtimes, that is, they are the same for all CPUs 31xC. Initially there is no difference between process and testing mode. Cycle extensions as a result of active testing and commissioning functions are listed in the table below.

Table 5-7 Cycle extension as a result of testing and commissioning functions

| Function | CPU 31xC |
|---|---|
| Monitor Variables | 50 µs per variable |
| Modify Variables | 50 µs per variable |
| Monitor block | 200 µs per monitored line |

### Parameter configuration

In **RUN mode** the maximum permissible cycle load by communication is not specified in "Cycle load by communication", but rather in "Maximum permitted cycle time increase by testing functions in process mode ⇒".Thus, the configured time is monitored absolutely in RUN mode. If exceeded, data acquisition is stopped. This is how *STEP 7*, for example, limits data requests to a point before a loop ends. When running in **Testing mode**, the loop is processed completely in every cycle. This can significantly increase cycle time.

## 5.3 Response Time

### 5.3.1 Overview

#### Definition of Response Time

The response time is the time between detection of an input signal and modification of an associated output signal.

#### Fluctuation width

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

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

#### Factors

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

- Delay of the inputs and outputs of signal modules or of the integrated I/O.
- Additional DP cycle times in a PROFIBUS-DP network (only with CPUs 31xC-2 DP)
- Handling in the user program

#### You can find the delay times ...

- in the technical data of signal modules (Reference Manual *Module data*)
- for integrated I/Os in *Technical data of integrated I/O*

#### DP cycle times in a PROFIBUS-DP network

if you have configured your PROFIBUS-DP network with *STEP 7* , *STEP 7* calculates the normally to be expected DP cycle time. You can then view the DP cycle time for your configuration on the PG.

The following figure gives you an overview of the DP cycle time. In this example, we assume that each DP slave has an average of 4 bytes of data.

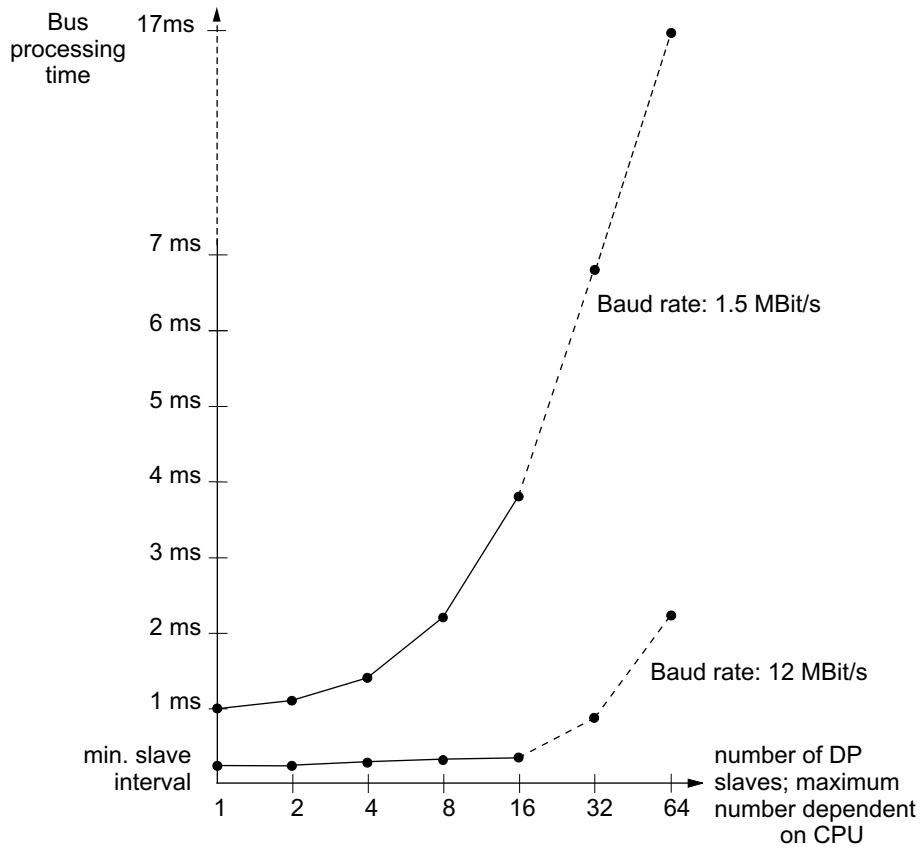



Figure 5-7 DP Cycle Times in the PROFIBUS-DP Network

With multi-master operation on a PROFIBUS-DP network you must consider the DP cycle time for each individual master. That is, you will have to calculate the times for each master separately and then add up the results.

**See also:** *Longest Response Time/ Shortest Response Time*

## 5.3.2 Shortest Response Time

### Conditions for Shortest Response Time

The figure below shows you the conditions under which the shortest response time is reached.

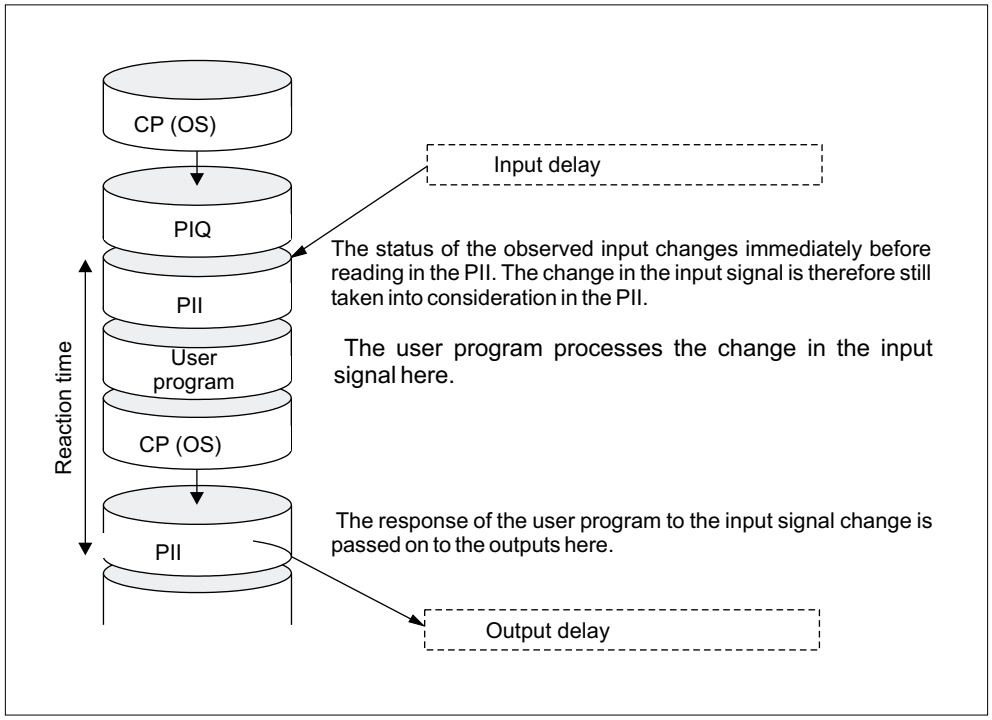



Figure 5-8 Shortest Response Time

### Calculation

The (shortest) response time consists of the following:

- 1 x Process image transfer time for the inputs +
- 1 x Process image transfer time for the outputs +
- 1 x Program execution time +
- 1 × Operating system processing time at the SCC +
- I/O delay

This corresponds to the total cycle time plus input and output delay.

## 5.3.3 Longest Response Time

### Conditions for Longest Response Time

The figure below shows the conditions that result in the longest response time.

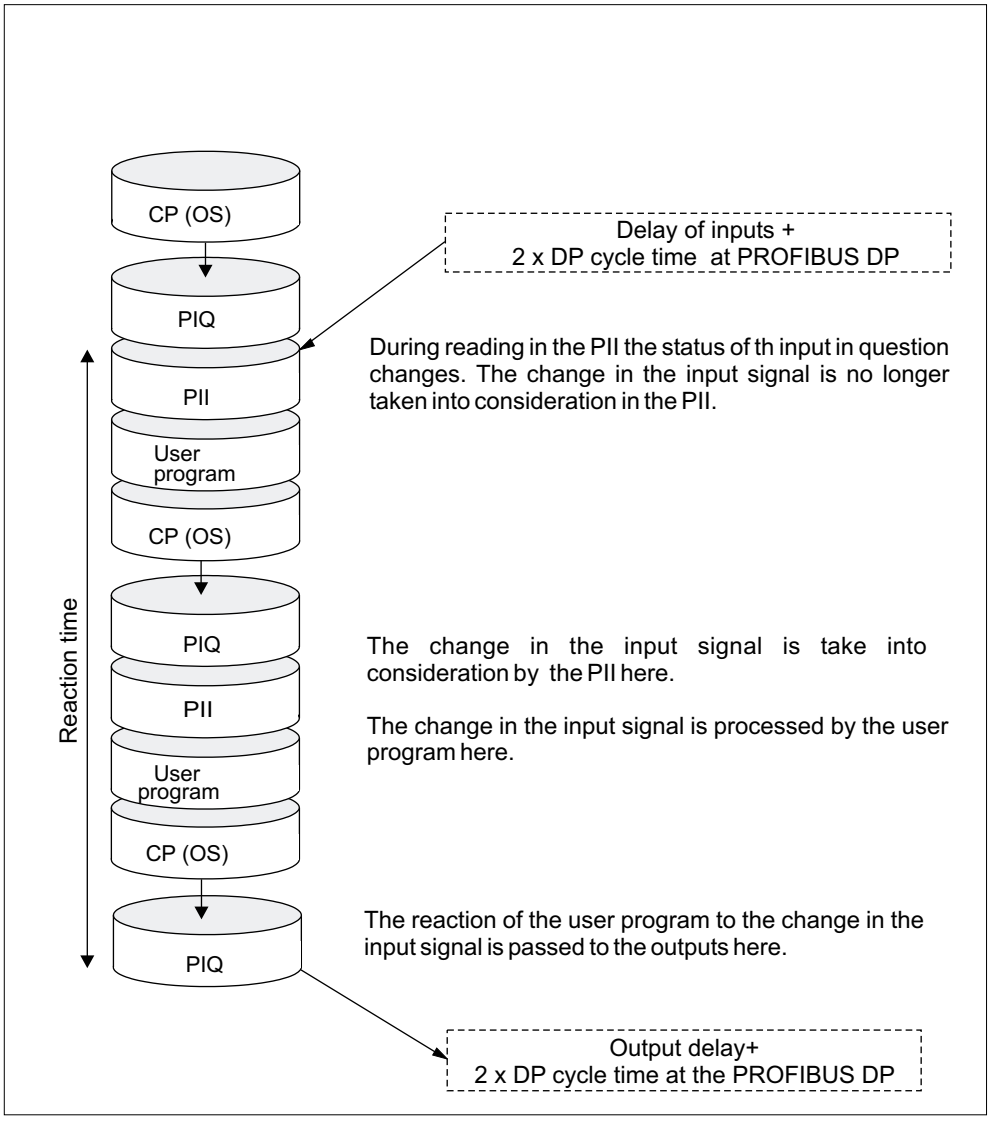



Figure 5-9 Longest Response Time

### Calculation

The (longest) response time consists of the following:

- 2 x Process image transfer time for the inputs +
- 2 x Process image transfer time for the outputs +
- 2 x Operating system execution time +
- 2 x Program execution time +
- 4 x the runtime of DP Slave message frames (includes processing in the DP Master) +
- I/O delay

Equivalent to twice the cycle time + input and output delay + twice the bus runtime.

**See also:** *How To Calculate Cycle/ Response Time*

*Reducing Response Time by Accessing the I/O*

## 5.3.4 Reducing Response Time by Accessing the I/O

### Reducing Response Time

You can gain faster response times with direct access to the I/O in your user program. For example, with

- L PIB or
- T PQW

you can partially avoid the response times described above.

---

**Note**

Faster response times are also possible by using process interrupts. Refer to the following chapters.

---

## 5.4 How to Calculate Cycle/ Response Time

### Introduction

This chapter gives you an overview of how to calculate the cycle/response time.

All tables are found in Chapter *Calculating the Cycle Time*.

### Cycle Time

1. Determine your user program's runtime with the help of the *Instruction list*.
2. Multiply the calculated value by the CPU-specific factor from the table *Extending user program execution time*.
3. Calculate and add the process image transfer time. Corresponding guide values are found in table *Data for calculating process image transfer time*.
4. Add the execution time at the scan cycle checkpoint. Corresponding guide values are found in the table *Operating system execution time at the scan cycle checkpoint*.
5. Include extensions caused by testing and commissioning functions in your calculation. You can find these values in the table *Cycle time extension with testing and commissioning functions. The final result is the*
6. **Cycle time***.*

### Extension of the Cycle Time, Caused by Interrupts and Communication

$$\frac{100}{100 - \text{“projected Communication load in \%”}}$$

Figure 5-10 Formula for calculating communication load

1. Multiply the cycle time by the factor as in the formula above.
2. Calculate the runtime of interrupt-processing program sections with the help of the instruction list. Add the corresponding value from Chapter *Calculating the Cycle Time*, Table *Extension of the cycle time as a result of nested interrupts.*
3. Multiply both values by the CPU-specific factor for extension of the user program execution time (see Table 5-3).
4. Add the value of the interrupt-processing program sequences to the theoretical cycle time, multiplied by the number of occurring/expected interrupts during the cycle time. The result is an approximation of the **physical cycle time**. Note down the result.

### Response Time

Table 5-8 Calculating the Response Time

| Shortest Response Time | Longest Response Time |
|---|---|
| - | Multiply the physical response time by the factor 2. |
| Include I/O delay. | Now add the calculated I/O delay and the DP cycle times in the PROFIBUS-DP network. |
| The result is the shortest response time. | The result is the longest response time. |

## 5.5 Interrupt Response Time

### 5.5.1 Overview

### Definition of the Interrupt Response Time

The interrupt response time is the time that expires between initial occurrence of an interrupt signal and the calling of the first instruction in the interrupt OB. The following rule generally applies: high-priority interrupts are executed first. This means the interrupt response time is increased by the program processing time of the higher priority interrupt OBs and the interrupt OBs of equal priority that have not yet been executed (queued).

### Calculation

The formulas below show you how you can calculate minimum and maximum interrupt response times.

Minimum CPU interrupt response time
+ minimum interrupt response time of signal modules
+ DP cycle time at PROFIBUS DP

= **shortest interrupt response time**

Maximum CPU interrupt response time
+ maximum interrupt response time of signal modules
+ 2 x DP cycle time at PROFIBUS DP

The maximum interrrupt response time is extended when the communications functions are active.The extension is calculated using the following formula:
CPU 31xC, tv = 200 $\mu$s + 1000 $\mu$s x n%,
Considerabe extension is possible with
n = cycle load by communication

Figure 5-11 Formula for calculating the interrupt response time

## Process/Diagnostic Interrupt Response Times of the CPUs

Table 5-9 Process/Diagnostic interrupt response times

| | Process interrupt response times | | | Diagnostic interrupt response times | |
|---|---|---|---|---|---|
| **CPU** | **external min.** | **external max.** | **Integrated I/O max.** | **Min.** | **Max.** |
| CPU 312C | 0.5 ms | 0.8 ms | 0.6 ms | 0.5 ms | 1.0 ms |
| CPU 313C | 0.4 ms | 0.6 ms | 0.5 ms | 0.4 ms | 1.0 ms |
| CPU 313C-2 | 0.4 ms | 0.7 ms | 0.5 ms | 0.4 ms | 1.0 ms |
| CPU 314C-2 | 0.4 ms | 0.7 ms | 0.5 ms | 0.4 ms | 1.0 ms |

## Signal Modules

The **Process Interrupt Response Time** of signal modules is composed of the following elements:

- Digital input modules

  Process interrupt response time = internal interrupt preparation time + input delay

  You will find the times in the data sheet for the individual analog input module.

- Analog input modules

  Process interrupt response time = internal interrupt preparation time + input delay

  The internal interrupt preparation time for the analog input modules is negligible. The conversion times can be found in the data sheet for the individual digital input modules.

The **Diagnostic Interrupt Response Time** of signal modules represents the time expiring between detection of a diagnostic event and triggering of the diagnostic interrupt by the signal module. This time is negligible.

## Process interrupt handling

Process interrupt handling begins when process interrupt OB40 is called. Higher priority interrupts cause the process interrupt handling routine to be interrupted. Direct accesses to the I/O are made at the execution time of the instruction. When the process interrupt handling routine has finished, either cyclic program execution continues or further interrupt OBs of equal or lower priority are called and executed.

## 5.5.2 Reproducibility of Delay/Watchdog Interrupts

### Definition of "Reproducibility"

**Delay Interrupt**:

The interval between the call of the first instruction in the OB and the programmed time of the interrupt.

**Watchdog Interrupt**:

The fluctuation width of the interval between two subsequent calls, respectively measured between the initial instruction of the interrupt OBs.

### Reproducibility

The following table shows the reproducibility of delay/watchdog interrupts of the CPUs.

Table 5-10 Reproducibility of the Delay and Watchdog Interrupts of the CPUs

| CPU | Delay Interrupt | Watchdog Interrupt |
|---|---|---|
| CPU 312C | +/- 200 µs | +/- 200 µs |
| CPU 313C | +/- 200 µs | +/- 200 µs |
| CPU 313C-2 | +/- 200 µs | +/- 200 µs |
| CPU 314C-2 | +/- 200 µs | +/- 200 µs |

These times only apply if the interrupt can actually be executed at this point and if it is not interrupted, for example, by higher-priority interrupts or interrupts of equal priority that have not yet been executed.

## 5.6 Sample calculations

### 5.6.1 Example of cycle time calculation

#### Assembly

You have assembled an S7 300 with the following modules in rack "0":

- a CPU 314C-2
- 2 Digital input modules SM 321; DI 32 x DC 24 V (4 bytes each in the PA)
- 2 Digital output modules SM 322; DO 32 x DC 24 V/0.5 A (4 byte each in the PA)

#### User program

According to the Instruction List, the user program has a runtime of 5 ms. There is no communication.

#### Calculating the cycle time

In this example, the cycle time is calculated from the following times:

- User program processing time:

  approx. 5 ms CPU specific factor 1.10 = approx. 5.5 ms
- Process image transfer time

  Input process image: 100 µs + 8 bytes x 37 µs = approx. 0.4 ms

  Output process image: 100 µs + 8 bytes x 37 µs = approx. 0.4 ms
- Operating system runtime at the scan cycle checkpoint:

  approx. 0.5 ms

**Cycle time** = 5.5 ms + 0.4 ms + 0.4 ms + 0.5 ms = 6.8 ms.

#### Calculating the actual cycle time

- There is no communication.
- No interrupts are being executed.

Hence, **the actual cycle time** is 6 ms.

### Calculating the longest response time

Longest Response Time

6.8 ms x 2 = 13.6 ms.

- I/O delay can be neglected.
- Since all modules are inserted in module rack 0, DP cycle times must not be taken into account.
- No interrupts are being executed.

## 5.6.2 Sample of response time calculation

### Assembly

You have assembled an S7 300 with the following modules in two racks:

- a CPU 314C-2

  Configuring cycle load as a result of communication: 40 %
- 4 Digital input modules SM 321; DI 32 x DC 24 V (4 bytes each in the PA)
- 3 Digital output modules SM 322; DO 16 x DC 24 V/0.5 A (2 bytes each in the PA)
- 2 Analog input modules SM 331; AI 8 x 12-bit (not in the PA)
- 2 Analog output modules SM 332; AO 4 x 12-bit (not in the PA)

### User program

According to the instruction list, the user program runtime is 10.0 ms.

### Calculating the cycle time

In this example, the cycle time is calculated from the following times:

- User program processing time:

  approx. 10 ms x CPU specific factor 1.10 = approx. 11 ms
- Process image transfer time

  Input process image: 100 µs + 16 bytes x 37 µs = approx. 0.7 ms

  Output process image: 100 µs + 6 bytes x 37 µs = approx. 0.3 ms
- Operating system runtime at the scan cycle checkpoint:

  approx. 0.5 ms

The cycle time is the sum total of the listed times:

**Cycle time** = 11.0 ms + 0.7 ms + 0.3 ms + 0.5 ms = 12.5 ms.

## Calculating the actual cycle time

Taking communication load into account:

12.5 ms * 100 / (100-40) = 20.8 ms.

Therefore, taking time-sharing into account, the **actual cycle time** is **21 ms**.

## Calculating the longest response time

- Longest response time = 21 ms * 2 = 42 ms.
- Delay times of the inputs and outputs
  - Maximum input delay of the digital module SM 321; DI 32 x DC 24 V is **4.8 ms** per channel.
  - The output delay of the digital output module SM 322; DO 16 x DC 24 V/0.5 A **can be neglected**.
  - Analog input module SM 331; AI 8 x 12-bit was configured for 50 Hz interference suppression. This yields a conversion time of 22 ms per channel. Since 8 channels are active, the result is a cycle time of **176 ms** for the analog input module.
  - Analog output module SM 332; AO 4 x 12-bit was configured for a measurement range of 0 ... 10 V. The result is a conversion time of 0.8 ms per channel. Since 4 channels are active, a cycle time of 3.2 ms is obtained. A settling time of 0.1 ms for a resistive load must be added to this value. The result is a response time of **3.3 ms** for an analog output.
- Since all modules are inserted in the master module rack, DP cycle times must not be taken into account.
- Response times with I/O delay times:
  - **Case 1:** An output channel of the digital output module is set when a digital input signal is read in. This results in a response time of:

    Response time = 42 ms + 4.8 ms = 46.8 ms.
  - **Case 2:** An analog value is read in and an analog value is output. This results in a response time of:

    **Longest response time** = 42 ms + 176 ms + 3.3 ms = 221.3 ms.

## 5.6.3 Example of Interrupt Response Time Calculation

### Assembly

You have assembled an S7-300, consisting of one CPU 314C-2 and four digital modules in the master rack. On of the digital input modules is the SM 321; DI 16 x DC 24 V; with process/diagnostic interrupt function.

You have enabled only the process interrupt in your CPU and SM parameter configuration. You decided not to use time controlled processing, diagnostics or error handling. You have configured a 20% communication load of the cycle.

You have configure an input delay of o.5 ms for the digital input module.

No activities are required at the scan cycle checkpoint.

### Calculation

In this example, the process interrupt response time is the result of the following time factors:

- Process interrupt response time of CPU 314C-2: approx. 0.7 ms
- Extension by communication load according to the formula:

  200 s + 1000 µs x 20 % = 400 s = 0.4 ms
- Process interrupt response time of SM 321; DI 16 x DC 24 V:
  - internal interrupt preparation time: 0.25 ms
  - Input delay: 0.5 ms
- Since the signal modules are inserted in the central rack, DP cycle times on the PROFIBUS-DP are irrelevant.

The process interrupt response time is the sum total of the specified time factors:

**Process interrupt response time** = 0.7 ms + 0.4 ms + 0.25 ms + 0.5 ms = **1.85** ms.

This calculated process interrupt response time expires from the time a signal is received at the digital input until the first instruction in OB 40.

# Technical Data of CPUs 31xC

# 6

## 6.1 CPU 312C

### Technical Data

| Technical Data | |
|---|---|
| **CPU and Product Version** | |
| Order number | 6ES7 312-5BD00-0AB0 |
| • Hardware version | 01 |
| • Firmware version | V1.0.0 |
| • Corresponding programming package | STEP 7 as of V 5.1 + SP 2 |
| **Memory** | |
| Work memory | |
| • Integrated | 16 KB |
| • Expandable | No |
| Load memory | pluggable (MMC) |
| Backup | ensured with MMC (maintenance-free) |
| **Processing times** | |
| Processing times for | |
| • Bit operation | min. 0. µs |
| • Word instructions | min. 0.4 µs |
| • Fixed-point mathematics | min. 5 µs |
| • Floating-point maths | min. 40 µs |
| **Timers/Counters and their retentivity** | |
| S7 counters | 128 |
| • Retentivity | adjustable |
| • Default | from C 0 to C 7 |
| • Counting range | 0 to 999 |
| IEC Counters | Yes |
| • Type | SFB |
| • Number | unlimited (limitation only by work memory) |
| S7 timers | 128 |

| Technical Data | |
|---|---|
| • Retentivity | adjustable |
| • Default | No retentivity |
| • Timing range | 10 ms to 9990 s |
| IEC Timers | Yes |
| • Type | SFB |
| • Number | unlimited (limitation only by work memory) |
| **Data areas and their retentive characteristics** | |
| Total retentive date area (including memory bits; timers; counters) | all |
| Bit memories | 128 bytes |
| • Retentivity | adjustable |
| • Retentivity is default setting | MB 0 to MB 15 |
| Clock memories | 8 (1 memory byte) |
| Data blocks | max. 63 |
| • Size | max. 16 KB |
| Local data per priority class | max. 256 bytes |
| **Blocks** | |
| Obs | See Instruction List |
| • Size | max. 16 KB |
| Nesting depth | |
| • Per priority class | 8 |
| • additionally within an error OB | 4 |
| FBs | max. 64 |
| • Size | max. 16 KB |
| FCs | max. 64 |
| • Size | max. 16 KB |
| **Address areas (I/Os)** | |
| Total I/O address area | max. 1024 bytes/1024 bytes (can be freely addressed) |
| I/O process image | 128 bytes/128 bytes |
| Digital channels | max. 256 |
| • Of those centralized | max. 256 |
| • integrated channels | 10 DI / 6 DO |
| Analog channels | max. 64 |
| • of those centralized | max. 64 |
| • integrated channels | None |
| **Assembly** | |
| Rack | max. 1 |
| Modules per module rack | max. 8 |
| Number of DP masters | |
| • Integrated | None |

| Technical Data | |
|---|---|
| • Via CP | max. 1 |
| Function modules and communication processors which can be operated | |
| • FM | max. 8 |
| • CP (PtP) | max. 8 |
| • CP (LAN) | max. 4 |
| **Time-of-day** | |
| Real time clock | yes (SW clock) |
| • Buffered | No |
| • Accuracy | Deviation per day < 10 s |
| Operating hours counter | 1 |
| • Number | 0 |
| • Range of values | 0 to 32767 hours |
| • Selectivity | 1 hour |
| • Retentive | yes; requires restarting at every restart |
| Clock synchronisation | Yes |
| • In the PLC | Master |
| • on MPI | Master/Slave |
| **S7 message functions** | |
| Number of stations which can log in for message functions (e.g. OS) | max. 3 |
| Process diagnostic messages | Yes |
| • simultaneously active interrupt S blocks | max. 20 |
| **Testing and commissioning functions** | |
| Status/Modify Variables | Yes |
| • Variable | Inputs, outputs, flags, DBs, timers, counters |
| • Number of variables | max. 30 |
| Of those as status variable | max. 30 |
| Of those as control variable | max. 14 |
| Force | Yes |
| • Variable | Inputs, outputs |
| • Number of variables | max. 10 |
| Monitor block | Yes |
| Single sequence | Yes |
| Breakpoint | 2 |
| Diagnostic buffer | Yes |
| • Number of entries (not configurable) | max. 100 |
| **Communication functions** | |
| PG/OP communication | Yes |
| Global data communication | Yes |
| • Number of GD packets | max. 4 |

| Technical Data | |
|---|---|
| Sending station | max. 4 |
| Receiving station | max. 4 |
| • Size of GD packets | max. 22 bytes |
| Of those are consistent | 22 bytes |
| S7 basic communication | Yes |
| • User data per job | max. 76 bytes |
| Of those are consistent | 32 bytes (with XPUT/XGET) |
| S7 communication | |
| • As Server | Yes |
| • User data per job | max. 180 bytes (with PUT/GET) |
| Of those are consistent | 32 bytes |
| S5-compatible communication | No |
| Standard communication | No |
| Number of connections | max. 6 |
| Usable for | |
| • PG communication | max. 5 |
| Reserved (Default) | 1 |
| Adjustable | from 1 to 5 |
| • OP communication | max. 5 |
| Reserved (Default) | 1 |
| Adjustable | from 1 to 5 |
| • S7 basic communication | max. 2 |
| Reserved (Default) | 2 |
| Adjustable | from 0 to 2 |
| Routing | No |
| **Interfaces** | |
| **1st interface** | |
| Type of interface | integrated RS485 interface |
| Physics | RS485 |
| Galvanically isolated | No |
| Interface current supply (15 to 30 V DC) | max. 200 mA |
| **Functionality** | |
| • MPI | Yes |
| • PROFIBUS-DP | No |
| • Point-to-point communication | No |
| **MPI** | |
| Number of connections | 6 |
| Services | |
| • PG/OP communication | Yes |
| • Routing | No |

| **Technical Data** | |
|---|---|
| • Global data communication | Yes |
| • S7 basic communication | Yes |
| • S7 communication | |
| As Server | Yes |
| As Client | No |
| • Transmission rates | max. 187.5 Kbps |
| **Programming** | |
| Programming language | LAD/FBD/STL |
| Stored instructions | See Instruction List |
| Nesting levels | 8 |
| System functions (SFCs) | See Instruction List |
| System function blocks (SFBs) | See Instruction List |
| User program security | Yes |
| **Integrated I/O** | |
| • Default addresses of the integrated | |
| Digital inputs | 124.0 to 125.1 |
| Digital outputs | 124.0 to 124.5 |
| **Integrated functions** | |
| Counter | 2 Channels (see the Manual *Technological Functions*) |
| Frequency meter | 2 channels, up to max. 10 kHz (see the Manual *Technological Functions*) |
| Pulse outputs | 2 channels for pulse width modulation, up to max. 2.5 kHz (see the Manual *Technological Functions*) |
| Controlled Positioning | No |
| Integrated SFB "Controlling" | No |
| **Dimensions** | |
| Mounting dimensions W x H x D (mm) | 80 x 125 x 130 |
| Weight | 409 g |
| **Voltages, Currents** | |
| Power supply (nominal value) | 24V DC |
| • Permissible range | 20.4 V to 28.8 V |
| Current consumption (no-load operation) | normally 60 mA |
| Inrush current | normally 11A |
| $I^2t$ | 0.7 $A^2s$ |
| External fusing for supply lines (recommendation) | LS switch Type C min. 2 A,<br>LS switch Type B min. 4 A |
| Power losses | normally 6 W |

### Cross-reference

In Chapter *Technical data of the integrated I/O* you can find

- the technical data of integrated I/Os, under *Digital inputs of CPUs 31xC* and *Digital outputs of CPUs 31xC*.
- the block diagrams of the integrated I/Os, under *Arrangement and usage of the integrated I/Os*.

## 6.2 CPU 313C

### Technical Data

| Technical Data | |
|---|---|
| **CPU and Product Version** | |
| Item number | 6ES7 313-5BE00-0AB0 |
| • Hardware version | 01 |
| • Firmware version | V1.0.0 |
| • Corresponding programming package | STEP 7 as of V 5.1 + SP 2 |
| **Memory** | |
| Work memory | |
| • Integrated | 32 KB |
| • Expandable | No |
| Load memory | pluggable (MMC) |
| Backup | ensured with MMC (maintenance-free) |
| **Processing times** | |
| Processing times for | |
| • Bit operation | min. 0.1 µs |
| • Word instructions | min. 0. µs |
| • Fixed-point mathematics | min. 2 µs |
| • Floating-point maths | min. 20 µs |
| **Timers/Counters and their retentivity** | |
| S7 counters | 256 |
| • Retentivity | Adjustable |
| • Default | from C 0 to C 7 |
| • Counting range | 0 to 999 |
| IEC Counters | Yes |
| • Type | SFB |
| • Number | unlimited (limitation only by work memory) |
| S7 timers | 256 |
| • Retentivity | Adjustable |
| • Default | No retentivity |
| • Timing range | 10 ms to 9990 s |
| IEC Timers | Yes |

| Technical Data | |
|---|---|
| • Type | SFB |
| • Number | unlimited (limitation only by work memory) |
| **Data areas and their retentive characteristics** | |
| Total retentive date area (including memory bits; timers; counters) | all |
| Bit memories | 256 bytes |
| • Retentivity | Adjustable |
| • Retentivity is default setting | MB 0 to MB 15 |
| Clock memories | 8 (1 memory byte) |
| Data blocks | max. 127 |
| • Size | max. 16 KB |
| Local data per priority class | max. 510 bytes |
| **Blocks** | |
| OBs | See Instruction List |
| • Size | max. 16 KB |
| Nesting depth | |
| • per priority class | 8 |
| • additionally within an error OB | 4 |
| FBs | max. 128 |
| • Size | max. 16 KB |
| FCs | max. 128 |
| • Size | max. 16 KB |
| **Address areas (I/Os)** | |
| Total I/O address area | max. 1024 bytes/1024 bytes (can be freely addressed) |
| I/O process image | 128 bytes/128 bytes |
| Digital channels | max. 1016 |
| • of those centralized | max. 992 |
| • integrated channels | 24 DI / 16 DO |
| Analog channels | max. 253 |
| • of those centralized | max. 248 |
| • integrated channels | 4 + 1 AI / 2 AO |
| **Assembly** | |
| Rack | max. 4 |
| Modules per module rack | max. 8; max. 7 in module rack 3 |
| Number of DP masters | |
| • Integrated | None |
| • via CP | max. 2 |
| Function modules and communication processors which can be operated | |
| • FM | max. 8 |
| • CP (PtP) | max. 8 |
| • CP (LAN) | max. 6 |
| **Time-of-day** | |
| Real time clock | yes (HW clock) |
| • Buffered | Yes |

| Technical Data | |
|---|---|
| • Buffering period | normally 6 weeks (at an ambient temperature of 40°C) |
| • Accuracy | Deviation per day < 10 s |
| Operating hours counter | 1 |
| • Number | 0 |
| • Range of values | 0 to 32767 hours |
| • Selectivity | 1 hour |
| • Retentive | yes; requires restarting at every restart |
| Clock synchronisation | Yes |
| • in the PLC | Master |
| • on MPI | Master/Slave |
| **S7 message functions** | |
| Number of stations which can log in for message functions (e.g. OS) | max. 5 |
| Process diagnostic messages | Yes |
| • simultaneously active interrupt S blocks | max. 20 |
| **Testing and commissioning functions** | |
| Status/Modify Variables | Yes |
| • Variable | Inputs, outputs, flags, DBs, timers, counters |
| • Number of variables | max. 30 |
| Of those as status variable | max. 30 |
| Of those as control variable | max. 14 |
| Force | Yes |
| • Variable | Inputs, outputs |
| • Number of variables | max. 10 |
| Monitor block | Yes |
| Single sequence | Yes |
| Breakpoint | 2 |
| Diagnostic buffer | Yes |
| • Number of entries (not configurable) | max. 100 |
| **Communication functions** | |
| PG/OP communication | Yes |
| Global data communication | Yes |
| • Number of GD packets | max. 4 |
| Sending station | max. 4 |
| Receiving station | max. 4 |
| • Size of GD packets | max. 22 bytes |
| Of those are consistent | 22 bytes |
| S7 basic communication | Yes |
| • User data per job | max. 76 bytes |
| Of those are consistent | 32 bytes (with XPUT/XGET) |
| S7 communication | |
| • as Server | Yes |
| • as Client | Yes (via CP and loadable FB) |
| • User data per job | max. 180 bytes (with PUT/GET) |
| Of those are consistent | 32 bytes |
| S5-compatible communication | No |

| Technical Data | |
|---|---|
| Standard communication | No |
| Number of connections | max. 8 |
| usable for | |
| • PG communication | max. 7 |
| Reserved (Default) | 1 |
| Adjustable | from 1 to 7 |
| • OP communication | max. 7 |
| Reserved (Default) | 1 |
| Adjustable | from 1 to 7 |
| • S7 basic communication | max. 4 |
| Reserved (Default) | 4 |
| Adjustable | from 0 to 4 |
| Routing | No |
| **Interfaces** | |
| **1st interface** | |
| Type of interface | integrated RS485 interface |
| Physics | RS485 |
| Galvanically isolated | No |
| Interface current supply (15 to 30 V DC) | max. 200 mA |
| **Functionality** | |
| • MPI | Yes |
| • PROFIBUS-DP | No |
| • PtP communication | No |
| **MPI** | |
| Number of connections | 8 |
| Services | |
| • PG/OP communication | Yes |
| • Routing | No |
| • Global data communication | Yes |
| • S7 basic communication | Yes |
| • S7 communication | |
| as Server | Yes |
| as Client | Yes (via CP and loadable FB) |
| • Transmission rates | max. 187.5 Kbps |
| **Programming** | |
| Programming language | LAD/FBD/STL |
| Stored instructions | See Instruction List |
| Nesting levels | 8 |
| System functions (SFCs) | See Instruction List |
| System function blocks (SFBs) | See Instruction List |
| User program security | Yes |
| **Integrated I/O** | |
| • Default addresses of the integrated | |
| digital inputs | 124.0 to 126.7 |
| Digital outputs | 124.0 to 125.7 |
| Analog inputs | 752 to 761 |

| Technical Data | |
|---|---|
| Analog outputs | 752 to 755 |
| **Integrated functions** | |
| Counter | 3 Channels (see the Manual *Technological Functions*) |
| Frequency meter | 3 channels, up to max. 30 kHz (see the Manual *Technological Functions*) |
| Pulse outputs | 3 channels for pulse width modulation, up to max. 2.5 kHz (see the Manual *Technological Functions*) |
| Controlled Positioning | No |
| Integrated SFB "Controlling" | PID controller (see the Manual *Technological Functions*) |
| **Dimensions** | |
| Mounting dimensions W x H x D (mm) | 120 x 125 x 130 |
| Weight | 660 g |
| **Voltages, Currents** | |
| Power supply (nominal value) | 24V DC |
| • Permissible range | 20.4 V to 28.8 V |
| Current consumption (no-load operation) | normally 150 mA |
| Inrush current | normally 11A |
| $I^2t$ | 0.7 $A^2s$ |
| External fusing for supply lines (recommendation) | LS switch Type C min. 2 A,<br>LS switch Type B min. 4 A |
| Power losses | normally 14 W |

## Cross-reference

In Chapter *Technical data of the integrated I/O* you can find

- the technical data of integrated I/O under *Digital inputs of CPUs 31xC*, *Digital outputs of CPUs 31xC, Analog inputs of CPUs 31xC* and *Analog outputs of CPUs 31xC*.
- the block diagrams of the integrated I/Os, under *Arrangement and usage of the integrated I/Os*.

## 6.3 CPU 313C-2 PtP and CPU 313C-2 DP

### Technical Data

| Technical Data | | |
|---|---|---|
| **CPU and Product Version** | **CPU 313C-2 PtP** | **CPU 313C-2 DP** |
| Item number | 6ES7 313-6BE00-0AB0 | 6ES7 313-6CE00-0AB0 |
| • Hardware version | 01 | 01 |
| • Firmware version | V1.0.0 | V1.0.0 |
| Matching programming package | STEP 7 as of V 5.1 + SP 2 | STEP 7 as of V 5.1 + SP 2 |
| **Memory** | **CPU 313C-2 PtP** | **CPU 313C-2 DP** |
| Work memory | | |
| • Integrated | 32 KB | |
| • Expandable | No | |
| Load memory | pluggable (MMC) | |
| Backup | ensured with MMC (maintenance-free) | |
| **Processing times** | **CPU 313C-2 PtP** | **CPU 313C-2 DP** |
| Processing times for | | |
| • Bit operation | min. 0.1 µs | |
| • Word instructions | min. 0.2 µs | |
| • Fixed-point mathematics | min. 2 µs | |
| • Floating-point maths | min. 20 µs | |
| **Timers/Counters and their retentivity** | **CPU 313C-2 PtP** | **CPU 313C-2 DP** |
| S7 counters | 256 | |
| • Retentivity | Adjustable | |
| • Default | from C 0 to C 7 | |
| • Counting range | 0 to 999 | |
| IEC Counters | Yes | |
| • Type | SFB | |
| • Number | unlimited (limitation only by work memory) | |
| S7 timers | 256 | |
| • Retentivity | Adjustable | |
| • Default | No retentivity | |
| • Timing range | 10 ms to 9990 s | |
| IEC Timers | Yes | |
| • Type | SFB | |
| • Number | unlimited (limitation only by work memory) | |

| Technical Data | | |
|---|---|---|
| **Data areas and their retentive characteristics** | **CPU 313C-2 PtP** | **CPU 313C-2 DP** |
| Total retentive data area (including memory bits; timers; counters) | all | |
| Bit memories | 256 bytes | |
| • Retentivity | Adjustable | |
| • Retentivity is default setting | MB 0 to MB 15 | |
| Clock memories | 8 (1 memory byte) | |
| Data blocks | max. 127 | |
| • Size | max. 16 KB | |
| Local data per priority class | max. 510 bytes | |
| **Blocks** | **CPU 313C-2 PtP** | **CPU 313C-2 DP** |
| OBs | See Instruction List | |
| • Size | max. 16 KB | |
| Nesting depth | | |
| • per priority class | 8 | |
| • additionally within an error OB | 4 | |
| FBs | max. 128 | |
| • Size | max. 16 KB | |
| FCs | max. 128 | |
| • Size | max. 16 KB | |
| **Address areas (I/Os)** | **CPU 313C-2 PtP** | **CPU 313C-2 DP** |
| Total I/O address area | max. 1024 bytes/1024 bytes (can be freely addressed) | max. 1024 bytes/1024 bytes (can be freely addressed) |
| • Distributed | None | max. 1008 bytes |
| I/O process image | 128 bytes/128 bytes | 128 bytes/128 bytes |
| Digital channels | max. 1008 | max. 8192 |
| • of those centralized | max. 992 | max. 992 |
| • integrated channels | 16 DI / 16 DO | 16 DI / 16 DO |
| Analog channels | max. 248 | max. 512 |
| • of those centralized | max. 248 | max. 248 |
| • integrated channels | None | None |
| **Assembly** | **CPU 313C-2 PtP** | **CPU 313C-2 DP** |
| Rack | max. 4 | |
| Modules per module rack | max. 8; max. 7 in module rack 3 | |
| Number of DP masters | | |
| • Integrated | No | 1 |
| • via CP | max. 1 | max. 1 |
| Function modules and communication processors which can be operated | | |
| • FM | max. 8 | |

| Technical Data | | |
|---|---|---|
| • CP (PtP) | max. 8 | |
| • CP (LAN) | max. 6 | |
| **Time-of-day** | **CPU 313C-2 PtP** | **CPU 313C-2 DP** |
| Real time clock | yes (HW clock) | |
| • Buffered | Yes | |
| • Buffering period | normally 6 weeks (at an ambient temperature of 40°C) | |
| • Accuracy | Deviation per day < 10 s | |
| Operating hours counter | 1 | |
| • Number | 0 | |
| • Range of values | 0 to 32767 hours | |
| • Selectivity | 1 hour | |
| • Retentive | yes; requires restarting at every restart | |
| Clock synchronisation | Yes | |
| • in the PLC | Master | |
| • on MPI | Master/Slave | |
| **S7 message functions** | **CPU 313C-2 PtP** | **CPU 313C-2 DP** |
| Number of stations which can log in for message functions (e.g. OS) | max. 5 | |
| Process diagnostic messages | Yes | |
| • simultaneously active interrupt S blocks | max. 20 | |
| **Testing and commissioning functions** | **CPU 313C-2 PtP** | **CPU 313C-2 DP** |
| Status/Modify Variables | Yes | |
| • Variable | Inputs, outputs, flags, DBs, timers, counters | |
| • Number of variables | max. 30 | |
| Of those as status variable | max. 30 | |
| Of those as control variable | max. 14 | |
| Force | Yes | |
| • Variable | Inputs, outputs | |
| • Number of variables | max. 10 | |
| Monitor block | Yes | |
| Single sequence | Yes | |
| Breakpoint | 2 | |
| Diagnostic buffer | Yes | |
| • Number of entries (not configurable) | max. 100 | |
| **Communication functions** | **CPU 313C-2 PtP** | **CPU 313C-2 DP** |
| PG/OP communication | Yes | |
| Global data communication | Yes | |
| • Number of GD packets | max. 4 | |

| Technical Data | | |
|---|---|---|
| Sending station | max. 4 | |
| Receiving station | max. 4 | |
| • Size of GD packets | max. 22 bytes | |
| Of those are consistent | 22 bytes | |
| S7 basic communication | Yes (server) | |
| • User data per job | max. 76 bytes | |
| Of those are consistent | 32 bytes (with XPUT/XGET) | |
| S7 communication | | |
| • as Server | Yes | |
| • as Client | Yes (via CP and loadable FB) | |
| • User data per job | max. 180 bytes (with PUT/GET) | |
| Of those are consistent | 32 bytes | |
| S5-compatible communication | No | |
| Standard communication | No | |
| Number of connections | max. 8 | |
| usable for | | |
| • PG communication | max. 7 | |
| Reserved (Default) | 1 | |
| Adjustable | from 1 to 7 | |
| • OP communication | max. 7 | |
| Reserved (Default) | 1 | |
| Adjustable | from 1 to 7 | |
| • S7 basic communication | max. 4 | |
| Reserved (Default) | 4 | |
| Adjustable | from 0 to 4 | |
| Routing | No | max. 4 |
| **Interfaces** | **CPU 313C-2 PtP** | **CPU 313C-2 DP** |
| **1st interface** | | |
| Type of interface | integrated RS485 interface | |
| Physics | RS485 | |
| Galvanically isolated | No | |
| Interface current supply (15 to 30 V DC) | max. 200 mA | |
| **Functionality** | | |
| • MPI | Yes | |
| • PROFIBUS-DP | No | |
| • Point-to-point communication | No | |
| **MPI** | | |
| Number of connections | 8 | |
| Services | | |
| • PG/OP communication | Yes | |

| Technical Data | | |
|---|---|---|
| • Routing | No | Yes |
| • Global data communication | Yes | |
| • S7 basic communication | Yes | |
| • S7 communication | | |
| as Server | Yes | |
| as Client | Yes (via CP and loadable FB) | |
| • Transmission rates | max. 187.5 Kbps | |
| **2nd interface** | **CPU 313C-2 PtP** | **CPU 313C-2 DP** |
| Type of interface | integrated RS422/RS485 interface | integrated RS485 interface |
| Physics | RS 422/485 | RS485 |
| Galvanically isolated | Yes | Yes |
| Interface current supply (15 to 30 V DC) | No | max. 200 mA |
| Number of connections | None | 8 |
| **Functionality** | | |
| • MPI | No | No |
| • PROFIBUS-DP | No | Yes |
| • Point-to-point communication | Yes | No |
| **DP master** | | |
| Number of connections | – | 8 |
| Services | | |
| • PG/OP communication | – | Yes |
| • Routing | – | Yes |
| • Global data communication | – | No |
| • S7 basic communication | – | No |
| • S7 communication | – | No |
| • Equidistance | – | Yes |
| • SYNC/FREEZE | – | Yes |
| • Activation/deactivation of DP slaves | – | Yes |
| • Transmission rates | – | Up to 12 Mbps |
| • Number of DP Slaves per station | – | max. 32; |
| • Address area | – | max. 1 Kbyte I / 1 Kbyte O |
| • User data per DP slave | – | max. 244 bytes I / 244 bytes O |
| **DP Slave** | | |
| Number of connections | – | 8 |
| Services | | |
| • PG/OP communication | – | Yes |
| • Routing | – | No |

| Technical Data | | |
|---|---|---|
| • Global data communication | – | No |
| • S7 basic communication | – | No |
| • S7 communication | – | No |
| • Direct data exchange | – | Yes |
| • Transmission rates | – | Up to 12 Mbps |
| • Transfer memory | – | 244 bytes I / 244 bytes O |
| • Address areas | – | max. 32 with max. 32 bytes each |
| **Point-to-Point communication** | | |
| • Transmission rates | 38.4 Kbps half duplex<br>19.2 Kbps full duplex | – |
| • Cable length | max. 1200 m | – |
| • The user program can control the interface | Yes | – |
| • The interface can trigger a break or an interrupt in the user program | Yes (message with break ID) | – |
| • Protocol driver | 3964 (R); ASCII | – |
| **Programming** | **CPU 313C-2 PtP** | **CPU 313C-2 DP** |
| Programming language | LAD/FBD/STL | |
| Stored instructions | See Instruction List | |
| Nesting levels | 8 | |
| System functions (SFCs) | See Instruction List | |
| System function blocks (SFBs) | See Instruction List | |
| User program security | Yes | |
| **Integrated I/O** | **CPU 313C-2 PtP** | **CPU 313C-2 DP** |
| • Default addresses of the integrated | | |
| digital inputs | 124.0 to 125.7 | |
| Digital outputs | 124.0 to 125.7 | |
| **Integrated functions** | | |
| Counter | 3 Channels (see the Manual *Technological Functions*) | |
| Frequency meter | 3 channels, up to max. 30 kHz (see the Manual *Technological Functions*) | |
| Pulse outputs | 3 channels for pulse width modulation, up to max. 2.5 kHz (see the Manual *Technological Functions*) | |
| Controlled Positioning | No | |
| Integrated SFB "Controlling" | PID controller (see the Manual *Technological Functions*) | |
| **Dimensions** | **CPU 313C-2 PtP** | **CPU 313C-2 DP** |
| Mounting dimensions W x H x D (mm) | 120 x 125 x 130 | |
| Weight | approx. 566 g | |

| Technical Data | | |
|---|---|---|
| **Voltages, Currents** | **CPU 313C-2 PtP** | **CPU 313C-2 DP** |
| Power supply (nominal value) | 24V DC | |
| • Permissible range | 20.4 V to 28.8 V | |
| Current consumption (no-load operation) | normally 100 mA | |
| Inrush current | normally 11A | |
| $I^2t$ | 0.7 $A^2s$ | |
| External fusing for supply lines (recommendation) | LS switch Type C min. 2 A, Type B min. 4 A | |
| Power losses | normally 10 W | |
| **Standards and Approvals** | **CPU 313C-2 PtP** | **CPU 313C-2 DP** |
| PNO Certificate | | |
| • DP master | – | |
| • DP Slave | – | |

## Cross-reference

In Chapter *Technical data of the integrated I/O* you can find

- the technical data of integrated I/Os, under *Digital inputs of CPUs 31xC* and *Digital outputs of CPUs 31xC*.
- the block diagrams of the integrated I/Os, under *Arrangement and usage of the integrated I/Os*.

## 6.4 CPU 314C-2 PtP and CPU 314C-2 DP

### Technical Data

| Technical Data | | |
|---|---|---|
| **CPU and Product Version** | **CPU 314C-2 PtP** | **CPU 314C-2 DP** |
| Item number | 6ES7 314-6BF00-0AB0 | 6ES7 314-6CF00-0AB0 |
| • Hardware version | 01 | 01 |
| • Firmware version | V1.0.0 | V1.0.0 |
| Matching programming package | STEP 7 as of V 5.1 + SP 2 | STEP 7 as of V 5.1 + SP 2 |
| **Memory** | **CPU 314C-2 PtP** | **CPU 314C-2 DP** |
| Work memory | | |
| • Integrated | 48 KB | |
| • Expandable | No | |
| Load memory | pluggable (MMC) | |
| Backup | ensured with MMC (maintenance-free) | |
| **Processing times** | **CPU 314C-2 PtP** | **CPU 314C-2 DP** |
| Processing times for | | |
| • Bit operation | min. 0.1 µs | |
| • Word instructions | min. 0.2 µs | |
| • Fixed-point mathematics | min. 2 µs | |
| • Floating-point maths | min. 20 µs | |
| **Timers/Counters and their retentivity** | **CPU 314C-2 PtP** | **CPU 314C-2 DP** |
| S7 counters | 256 | |
| • Retentivity | Adjustable | |
| • Default | from C 0 to C 7 | |
| • Counting range | 0 to 999 | |
| IEC Counters | Yes | |
| • Type | SFB | |
| • Number | unlimited (limitation only by work memory) | |
| S7 timers | 256 | |
| • Retentivity | Adjustable | |
| • Default | No retentivity | |
| • Timing range | 10 ms to 9990 s | |
| IEC Timers | Yes | |
| • Type | SFB | |
| • Number | unlimited (limitation only by work memory) | |

| Technical Data | | |
|---|---|---|
| **Data areas and their retentive characteristics** | **CPU 314C-2 PtP** | **CPU 314C-2 DP** |
| Total retentive data area (including memory bits; timers; counters) | all | |
| Bit memories | 256 bytes | |
| • Retentivity | Adjustable | |
| • Retentivity is default setting | MB 0 to MB 15 | |
| Clock memories | 8 (1 memory byte) | |
| Data blocks | max. 127 | |
| • Size | max. 16 KB | |
| Local data per priority class | max. 510 bytes | |
| **Blocks** | **CPU 314C-2 PtP** | **CPU 314C-2 DP** |
| OBs | See Instruction List | |
| • Size | max. 16 KB | |
| Nesting depth | | |
| • per priority class | 8 | |
| • additionally within an error OB | 4 | |
| FBs | max. 128 | |
| • Size | max. 16 KB | |
| FCs | max. 128 | |
| • Size | max. 16 KB | |
| **Address areas (I/Os)** | **CPU 314C-2 PtP** | **CPU 314C-2 DP** |
| Total I/O address area | max. 1024 bytes/1024 bytes (can be freely addressed) | max. 1024 bytes/1024 bytes (can be freely addressed) |
| • Distributed | None | max. 1,000 bytes |
| I/O process image | 128 bytes/128 bytes | 128 bytes/128 bytes |
| Digital channels | max. 1016 | max. 8192 |
| • of those centralized | max. 992 | max. 992 |
| • integrated channels | 24 DI / 16 DO | 24 DI / 16 DO |
| Analog channels | max. 253 | max. 512 |
| • of those centralized | max. 248 | max. 248 |
| • integrated channels | 4 + 1 AI / 2 AO | 4 + 1 AI / 2 AO |
| **Assembly** | **CPU 314C-2 PtP** | **CPU 314C-2 DP** |
| Rack | max. 4 | |
| Modules per module rack | max. 8; max. 7 in module rack 3 | |
| Number of DP masters | | |
| • Integrated | No | 1 |
| • via CP | max. 1 | max. 1 |
| Function modules and communication processors which can be operated | | |

| Technical Data | | |
|---|---|---|
| • FM | max. 8 | |
| • CP (PtP) | max. 8 | |
| • CP (LAN) | max. 10 | |
| **Time-of-day** | **CPU 314C-2 PtP** | **CPU 314C-2 DP** |
| Real time clock | yes (HW clock) | |
| • Buffered | Yes | |
| • Buffering period | normally 6 weeks (at an ambient temperature of 40°C) | |
| • Accuracy | Deviation per day < 10 s | |
| Operating hours counter | 1 | |
| • Number | 0 | |
| • Range of values | 0 to 32767 hours | |
| • Selectivity | 1 hour | |
| • Retentive | yes; requires restarting at every restart | |
| Clock synchronisation | Yes | |
| • in the PLC | Master | |
| • on MPI | Master/Slave | |
| **S7 message functions** | **CPU 314C-2 PtP** | **CPU 314C-2 DP** |
| Number of stations which can log in for message functions (e.g. OS) | max. 7 | |
| Process diagnostic messages | Yes | |
| • simultaneously active interrupt S blocks | max. 20 | |
| **Testing and commissioning functions** | **CPU 314C-2 PtP** | **CPU 314C-2 DP** |
| Status/Modify Variables | Yes | |
| • Variable | Inputs, outputs, flags, DBs, timers, counters | |
| • Number of variables | max. 30 | |
| Of those as status variable | max. 30 | |
| Of those as control variable | max. 14 | |
| Force | Yes | |
| • Variable | Inputs, outputs | |
| • Number of variables | max. 10 | |
| Monitor block | Yes | |
| Single sequence | Yes | |
| Breakpoint | 2 | |
| Diagnostic buffer | Yes | |
| • Number of entries (not configurable) | max. 100 | |

| Technical Data | | |
|---|---|---|
| **Communication functions** | **CPU 314C-2 PtP** | **CPU 314C-2 DP** |
| PG/OP communication | Yes | |
| Global data communication | Yes | |
| • Number of GD packets | max. 4 | |
| Sending station | max. 4 | |
| Receiving station | max. 4 | |
| • Size of GD packets | max. 22 bytes | |
| Of those are consistent | 22 bytes | |
| S7 basic communication | Yes | |
| • User data per job | max. 76 bytes | |
| Of those are consistent | 32 bytes (with XPUT/XGET) | |
| S7 communication | | |
| • as Server | Yes | |
| • as Client | Yes (via CP and loadable FB) | |
| • User data per job | max. 180 bytes (with PUT/GET) | |
| Of those are consistent | 32 bytes | |
| S5-compatible communication | No | |
| Standard communication | No | |
| Number of connections | max. 12 | |
| usable for | | |
| • PG communication | max. 11 | |
| Reserved (Default) | 1 | |
| Adjustable | from 1 to 11 | |
| • OP communication | max. 11 | |
| Reserved (Default) | 1 | |
| Adjustable | from 1 to 11 | |
| • S7 basic communication | max. 8 | |
| Reserved (Default) | 8 | |
| Adjustable | from 0 to 8 | |
| Routing | No | max. 4 |
| **Interfaces** | **CPU 314C-2 PtP** | **CPU 314C-2 DP** |
| **1st interface** | | |
| Type of interface | integrated RS485 interface | |
| Physics | RS485 | |
| Galvanically isolated | No | |
| Interface current supply (15 to 30 V DC) | max. 200 mA | |
| **Functionality** | | |
| • MPI | Yes | |
| • PROFIBUS-DP | No | |
| • Point-to-point communication | No | |

| Technical Data | | |
|---|---|---|
| **MPI** | | |
| Number of connections | 12 | |
| Services | | |
| • PG/OP communication | Yes | |
| • Routing | No | Yes |
| • Global data communication | Yes | |
| • S7 basic communication | Yes | |
| • S7 communication | | |
| as Server | Yes | |
| as Client | Yes (via CP and loadable FB) | |
| • Transmission rates | max. 187.5 Kbps | |
| **2nd interface** | **CPU 314C-2 PtP** | **CPU 314C-2 DP** |
| Type of interface | integrated RS422/RS485 interface | integrated RS485 interface |
| Physics | RS 422/485 | RS485 |
| Galvanically isolated | Yes | Yes |
| Interface current supply (15 to 30 V DC) | No | max. 200 mA |
| Number of connections | None | 12 |
| **Functionality** | | |
| • MPI | No | No |
| • PROFIBUS-DP | No | Yes |
| • Point-to-point communication | Yes | No |
| **DP master** | | |
| Number of connections | – | 12 |
| Services | | |
| • PG/OP communication | – | Yes |
| • Routing | – | Yes |
| • Global data communication | – | No |
| • S7 basic communication | – | No |
| • S7 communication | – | No |
| • Equidistance | – | Yes |
| • SYNC/FREEZE | – | Yes |
| • Activation/deactivation of DP slaves | – | Yes |
| • Transmission rates | – | Up to 12 Mbps |
| • Number of DP Slaves per station | – | max. 32; |
| • Address area | – | max. 1 Kbyte I / 1 Kbyte O |
| • User data per DP slave | – | max. 244 bytes I / 244 bytes O |

| Technical Data | | |
|---|---|---|
| **DP Slave** | | |
| Number of connections | – | 12 |
| Services | | |
| • PG/OP communication | – | Yes |
| • Routing | – | No |
| • Global data communication | – | No |
| • S7 basic communication | – | No |
| • S7 communication | – | No |
| • Direct data exchange | – | Yes |
| • Transmission rates | – | Up to 12 Mbps |
| • Transfer memory | – | 244 bytes I / 244 bytes O |
| • Address areas | – | max. 32 with max. 32 bytes each |
| **Point-to-Point communication** | | |
| • Transmission rates | 38.4 Kbps half duplex<br>19.2 Kbps full duplex | – |
| • Cable length | max. 1200 m | – |
| • The user program can control the interface | Yes | – |
| • The interface can trigger a break or an interrupt in the user program | Yes (message with break ID) | – |
| • Protocol driver | 3964 (R); ASCII | – |
| **Programming** | **CPU 314C-2 PtP** | **CPU 314C-2 DP** |
| Programming language | LAD/FBD/STL | |
| Stored instructions | See Instruction List | |
| Nesting levels | 8 | |
| System functions (SFCs) | See Instruction List | |
| System function blocks (SFBs) | See Instruction List | |
| User program security | Yes | |
| **Integrated I/O** | **CPU 314C-2 PtP** | **CPU 314C-2 DP** |
| • Default addresses of the integrated | | |
| digital inputs | 124.0 to 126.7 | |
| Digital outputs | 124.0 to 125.7 | |
| Analog inputs | 752 to 761 | |
| Analog outputs | 752 to 755 | |

| Technical Data | | |
|---|---|---|
| **Integrated functions** | | |
| Counter | 4 Channels (see the Manual *Technological Functions*) | |
| Frequency meter | 4 channels, up to max. 60 kHz (see the Manual *Technological Functions*) | |
| Pulse outputs | 4 channels for pulse width modulation, up to max. 2.5 kHz (see the Manual *Technological Functions*) | |
| Controlled Positioning | 1 Channel (see the Manual *Technological Functions*) | |
| Integrated SFB "Controlling" | PID controller (see the Manual *Technological Functions*) | |
| **Dimensions** | **CPU 314C-2 PtP** | **CPU 314C-2 DP** |
| Mounting dimensions W x H x D (mm) | 120 x 125 x 130 | |
| Weight | approx. 676 g | |
| **Voltages, Currents** | **CPU 314C-2 PtP** | **CPU 314C-2 DP** |
| Power supply (nominal value) | 24V DC | |
| • Permissible range | 20.4 V to 28.8 V | |
| Current consumption (no-load operation) | normally 150 mA | |
| Inrush current | normally 11A | |
| $I^2t$ | 0.7 $A^2s$ | |
| External fusing for supply lines (recommendation) | LS switch Type C min. 2 A,<br>LS switch Type B min. 4 A | |
| Power losses | normally 14 W | |
| **Standards and Approvals** | **CPU 314C-2 PtP** | **CPU 314C-2 DP** |
| PNO Certificate | | |
| • DP master | – | |
| • DP Slave | – | |

## Cross-reference

In Chapter *Technical data of the integrated I/O* you can find

- the technical data of integrated I/O under *Digital inputs of CPUs 31xC*, *Digital outputs of CPUs 31xC, Analog inputs of CPUs 31xC* and *Analog outputs of CPUs 31xC*.
- the block diagrams of the integrated I/Os, under *Arrangement and usage of the integrated I/Os*.

# Technical Data of the Integrated I/O 7

## 7.1 Arrangement and Usage of Integrated I/Os

### Introduction

The integrated I/O of CPUs 31xC can be used for technological functions or as standard I/O.

The figures below illustrate possible usage of I/Os integrated in the CPUs.

**Further information on integrated I/O is found in the Manual "Technical Functions"**

### CPU 312C

| Standard | Input interrupt | Count | X1 | |
|---|---|---|---|---|
| | | | 1 ∅ | |
| DI | X | C0 (A) | 2 ∅ | DI+0.0 |
| DI | X | C0(B) | 3 ∅ | DI+0.1 |
| DI | X | C0 (N) | 4 ∅ | DI+0.2 |
| DI | X | Z1 (A) | 5 ∅ | DI+0.3 |
| DI | X | Z1 (B) | 6 ∅ | DI+0.4 |
| DI | X | Z1 (N) | 7 ∅ | DI+0.5 |
| DI | X | Sync 0 | 8 ∅ | DI+0.6 |
| DI | X | Sync 1 | 9 ∅ | DI+0.7 |
| DI | X | | 10 ∅ | DI+1.0 |
| DI | X | | 11 ∅ | DI+1.1 |
| | | | 12 ∅ | 2 M |
| | | | 13 ∅ | 1L+ |
| DO | | V0 | 14 ∅ | DO+0.0 |
| DO | | V1 | 15 ∅ | DO+0.1 |
| DO | | | 16 ∅ | DO+0.2 |
| DO | | | 17 ∅ | DO+0.3 |
| DO | | | 18 ∅ | DO+0.4 |
| DO | | | 19 ∅ | DO+0.5 |
| | | | 20 ∅ | 1 M |

| | |
|---|---|
| Cn | Counter n |
| A, B, N | Sensor signals |
| Sync n | Sync signal n (on-the-fly setting of actual value ) |
| Cn | Comparator n |
| X | Pin usable, if not occupied by technological functions. |

Figure 7-1 CPU 312C: Pin-out of the integrated DI/DO (Connector X1)

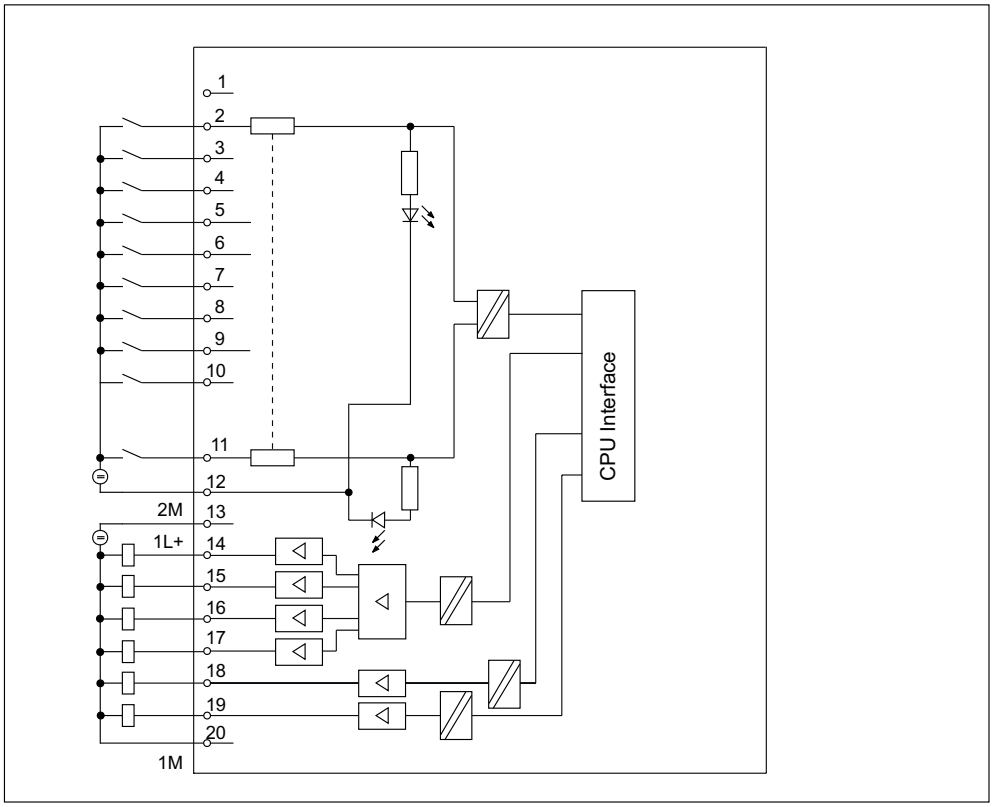



Figure 7-2 Basic Circuit Diagram of the Integrated Digital I/O or the CPU 312 C

## CPU 313C, CPU 313C-2 DP/PtP, CPU 314C-2 DP/PtP

| Standard DI | Input interrupt | Count | Posi-tioning 1) | X2 | | | | Positioning 1) | | Count | Standard DO |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | 1 ∅ | 1L+ | 2L+ | ∅ 21 | digital | analog | | |
| X | X | C0 (A) | A 0 | 2 ∅ | DI+0.0 | DO+0.0 | ∅ 22 | | | V0 | X |
| X | X | C0 (B) | B 0 | 3 ∅ | DI+0.1 | DO+0.1 | ∅ 23 | | | V1 | X |
| X | X | C0 (N) | N 0 | 4 ∅ | DI+0.2 | DO+0.2 | ∅ 24 | | | V2 | X |
| X | X | C1 (A) | Touch 0 | 5 ∅ | DI+0.3 | DO+0.3 | ∅ 25 | | | V3 1) | X |
| X | X | C1 (B) | Bero 0 | 6 ∅ | DI+0.4 | DO+0.4 | ∅ 26 | | | | X |
| X | X | C1 (N) | | 7 ∅ | DI+0.5 | DO+0.5 | ∅ 27 | | | | X |
| X | X | C2 (A) | | 8 ∅ | DI+0.6 | DO+0.6 | ∅ 28 | | Enable | | X |
| X | X | C2 (B) | | 9 ∅ | DI+0.7 | DO+0.7 | ∅ 29 | | | | X |
| | | | | 10 ∅ | | 2M | ∅ 30 | | | | |
| | | | | 11 ∅ | | 3L+ | ∅ 31 | | | | |
| X | X | C2 (N) | | 12 ∅ | DI+1.0 | DO+1.0 | ∅ 32 | R+ | | | X |
| X | X | C3 (A) | 1) | 13 ∅ | DI+1.1 | DO+1.1 | ∅ 33 | R- | | | X |
| X | X | C3 (B) | | 14 ∅ | DI+1.2 | DO+1.2 | ∅ 34 | Rapid | | | X |
| X | X | C3 (N) | | 15 ∅ | DI+1.3 | DO+1.3 | ∅ 35 | Creep | | | X |
| X | X | Sync 0 | | 16 ∅ | DI+1.4 | DO+1.4 | ∅ 36 | | | | X |
| X | X | Sync 1 | | 17 ∅ | DI+1.5 | DO+1.5 | ∅ 37 | | | | X |
| X | X | Sync 2 | | 18 ∅ | DI+1.6 | DO+1.6 | ∅ 38 | | | | X |
| X | X | Sync 3 1) | | 19 ∅ | DI+1.7 | DO+1.7 | ∅ 39 | | | | X |
| | | | | 20 ∅ | 1M | 3M | ∅ 40 | | | | |

| | |
|---|---|
| Cn | Counter n |
| A, B, N | Sensor signals |
| Sync n | Sync signal n (on-the-fly setting of actual value ) |
| Cn | Comparator n |
| Touch 0 | Touch probe 0 |
| Bero 0 | Reference-point switch 0 |
| R+, R- | Directional signal |
| Rapid | Rapid traverse |
| Creep | Creep speed |
| Enable | Enable Power Section |
| X | Pin usable if not occupied by technological functions. |

1) only CPU 314C-2

Figure 7-3 CPU 313C/313C-2/314C-2: Pin-out of the integrated DI/DO (Connector X2)

Details are found in the *Manual "Technical Functions, in Chapters "Counting", "Frequency Measurement" and "Pulse Width Modulation"*

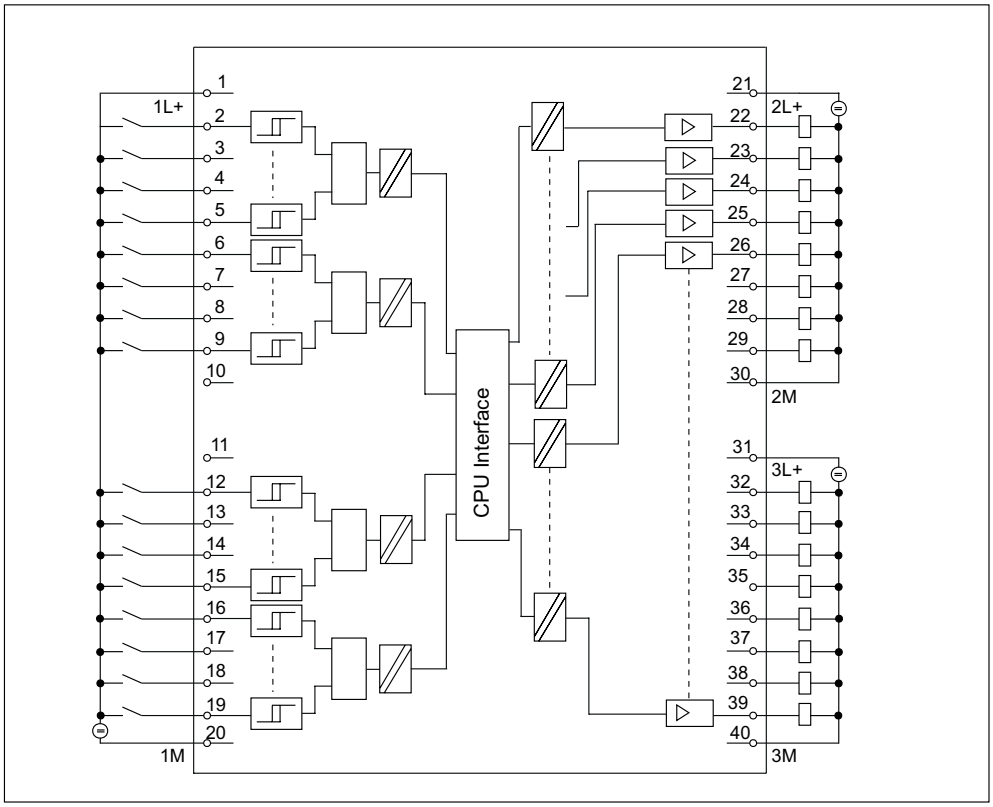



Figure 7-4 Basic Circuit Diagram of the Integrated Digital I/O or the CPUs 313C/313C-2/314C-2

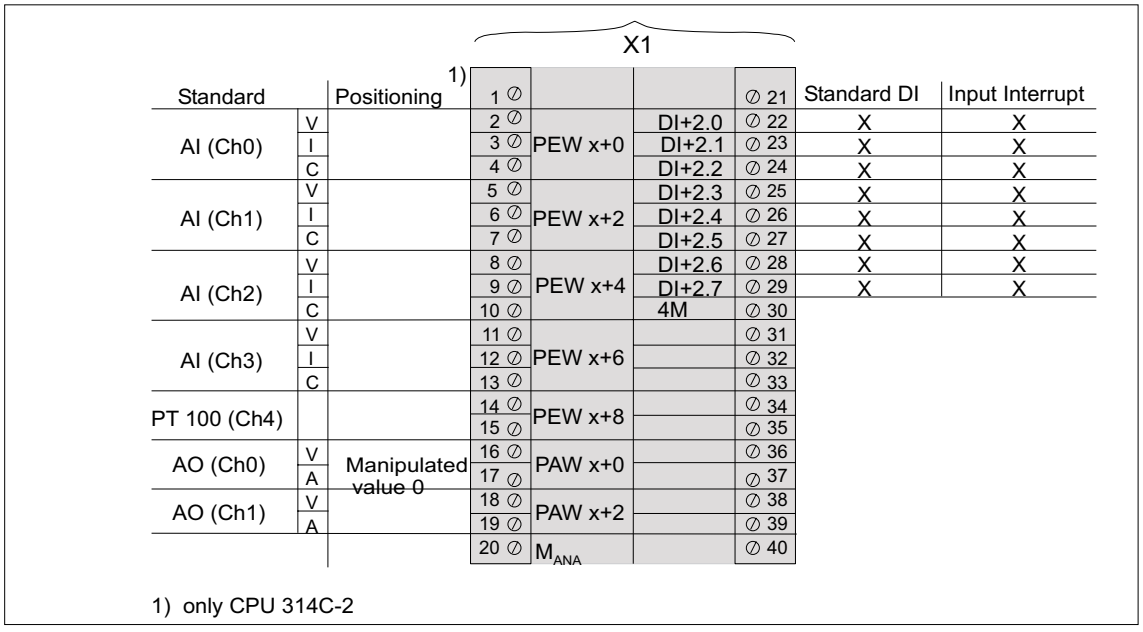



Figure 7-5 CPU 313C/314C-2: Pin-out of the integrated AI/AO and DI (Connector X1)

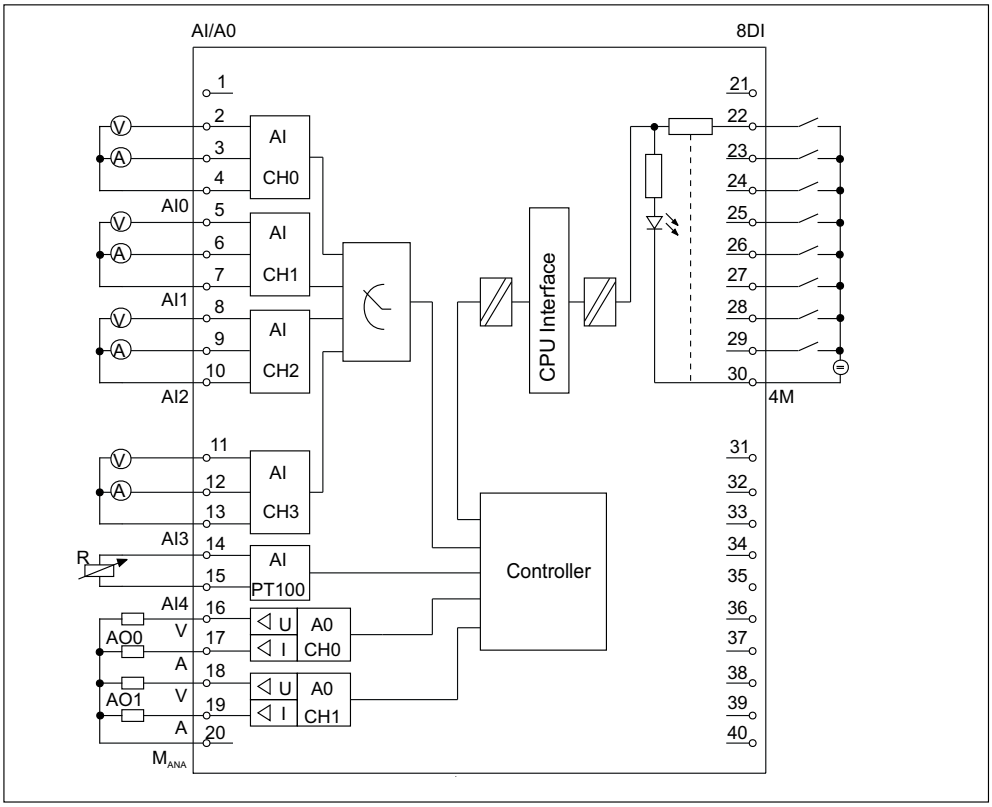



Figure 7-6 Basic Circuit Diagram of the Integrated Digital/Analog I/O of the CPUs 313C/314C-2

## Simultaneous use of technological functions and standard I/O

Technological functions and standard I/O can be used simultaneously with appropriate hardware. For example, you cannot use digital inputs as standard DI, if they are in use by counting functions.

Inputs used by technological functions can be accessed for reading. Outputs used by technological functions cannot be accessed for writing.

Possible effects on CPU performance are described in Chapter *Cycle/Response Times*.

## 7.2 Analog I/O

### Wiring of the Current/Voltage Inputs

The figure below shows the wiring diagram of the current/voltage inputs operated with 2-/4-wire measuring transducers.

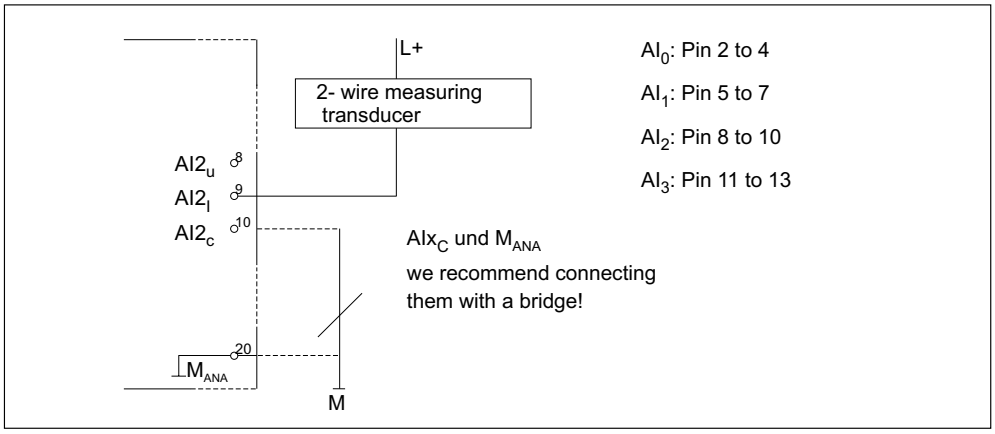



Figure 7-7 Wiring of an analog current/voltage input of CPU 313C/314C-2 with 2-wire measuring transducer

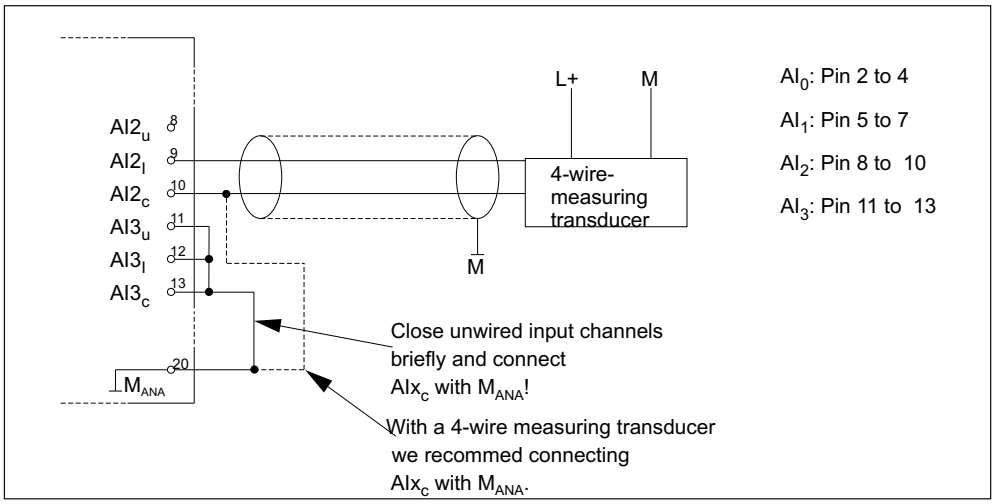



Figure 7-8 Wiring of an analog current/voltage input of CPU 313C/314C-2 with 4-wire measuring transducer

## Measuring principle

CPUs 31xC use the measurement principle of actual value encoding. Here they operate with a sampling rate of 1 kHz, that is, a new value is available at the input word register once every millisecond. This value can then be read via user program (e.g. L PEW). The "old" value is read again if the access times are lower than 1 ms.

## Input Filter (RC combination)

An integrated low-pass filter attenuates the analog input signals at channel 0 to 3. They are attenuated according to the curve in the figure below.

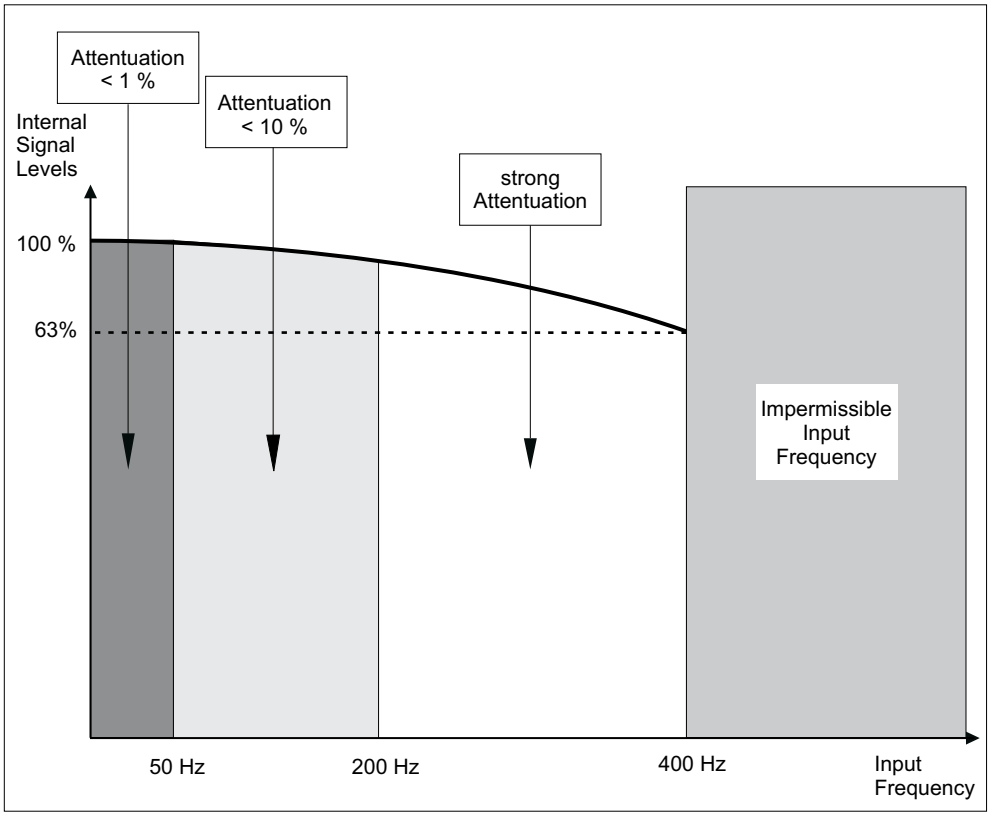



Figure 7-9 Conductive characteristics of the integrated low-pass filter

**Note**

The frequency of the input signal must not exceed 400 Hz.

### Input filter (Software filter)

The current/voltage inputs are equipped with a configurable software filter for the input signals. This software filter attenuates the configured interference frequencies (50/60 Hz) and multiples thereof.

The selected interference suppression also determines the integration time.
At an interference suppression of 50 Hz the software filter forms the average of the last 20 measurements and saves this result as measurement value.

**Note**

If the interference frequency is not 50/60 Hz or a multiple thereof, the input signal has to be filtered externally.
In this case frequency suppression must be configured with a value 400 Hz for this input. This equals a "Deactivation" of the software filter.

### Inputs not connected

The three inputs of a current/voltage analog output channel which are not connected should be shorted and connected to $M_{ANA}$ (Pin 20 of the front connector). This ensures maximum interference resistance for these analog inputs.

### Outputs not connected

In order to take analog outputs which are not in use off voltage, you must deactivate and leave them open in your configuration with *STEP 7*.

### Cross-reference

Details (e.g. display and processing of analog values) are found in Chapter 4 of the Reference Manual *Module Data*.

## 7.3 Configuration

### Introduction

You configure the integrated I/O of CPUs 31xC with *STEP 7*. Settings are always made when the CPU is in STOP mode. The parameters you created are downloaded from your PG to the S7-300 and written to CPU memory . You could also change the parameters with SFC55 in you user program (see the Reference Manual *System and Standard Functions*). Here, refer to the structure of data record 1 for the respective parameters.

### Parameters of Standard DI

The table below gives you an overview of the parameters for standard digital inputs.

Table 7-1 Parameters of Standard DI

| Parameters | Value range | Default setting | Efficiency range |
|---|---|---|---|
| Input delay (ms) | 0,1/0,5/3/15 | 3 | Channel Group |

The table below gives you an overview of the parameters for using digital inputs as interrupt inputs .

Table 7-2 Parameters of the Interrupt Inputs

| Parameters | Value range | Default setting | Efficiency range |
|---|---|---|---|
| Interrupt input | Disabled / rising edge | disabled | digital input |
| Interrupt input | Disabled / rising edge | disabled | digital input |

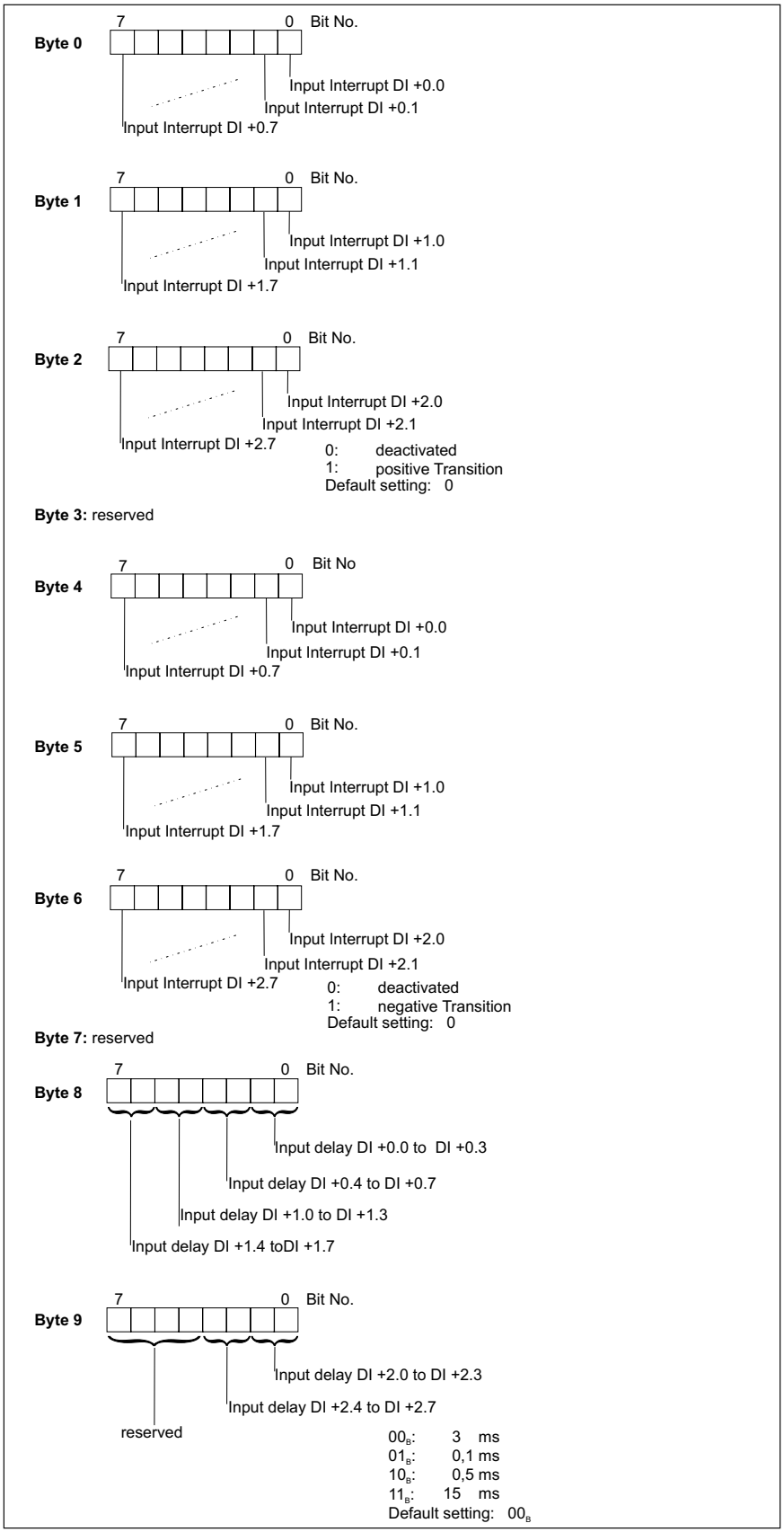

Byte 0
7
0
Bit No.
Input Interrupt DI +0.0
Input Interrupt DI +0.1
Input Interrupt DI +0.7
Byte 1
7
0
Bit No.
Input Interrupt DI +1.0
Input Interrupt DI +1.1
Input Interrupt DI +1.7
Byte 2
7
0
Bit No.
Input Interrupt DI +2.0
Input Interrupt DI +2.1
Input Interrupt DI +2.7
0: deactivated
1: positive Transition
Default setting: 0
Byte 3: reserved
Byte 4
7
0
Bit No
Input Interrupt DI +0.0
Input Interrupt DI +0.1
Input Interrupt DI +0.7
Byte 5
7
0
Bit No.
Input Interrupt DI +1.0
Input Interrupt DI +1.1
Input Interrupt DI +1.7
Byte 6
7
0
Bit No.
Input Interrupt DI +2.0
Input Interrupt DI +2.1
Input Interrupt DI +2.7
0: deactivated
1: negative Transition
Default setting: 0
Byte 7: reserved
Byte 8
7
0
Bit No.
Input delay DI +0.0 to DI +0.3
Input delay DI +0.4 to DI +0.7
Input delay DI +1.0 to DI +1.3
Input delay DI +1.4 toDI +1.7
Byte 9
7
0
Bit No.
Input delay DI +2.0 to DI +2.3
Input delay DI +2.4 to DI +2.7
reserved
00B: 3 ms
01B: 0,1 ms
10B: 0,5 ms
11B: 15 ms
Default setting: 00B

Figure 7-10 Structure of Data Record 1 for Standard DI and Interrupt Inputs (length is 10 bytes)

## Parameters of Standard DO

There are no parameters for standard digital outputs.

## Parameters of Standard AI

The following table gives you an overview of the parameters for standard analog inputs (see also Chapter 4.3 in the Reference Manual *Module Data*).

Table 7-3 Parameters of Standard AI

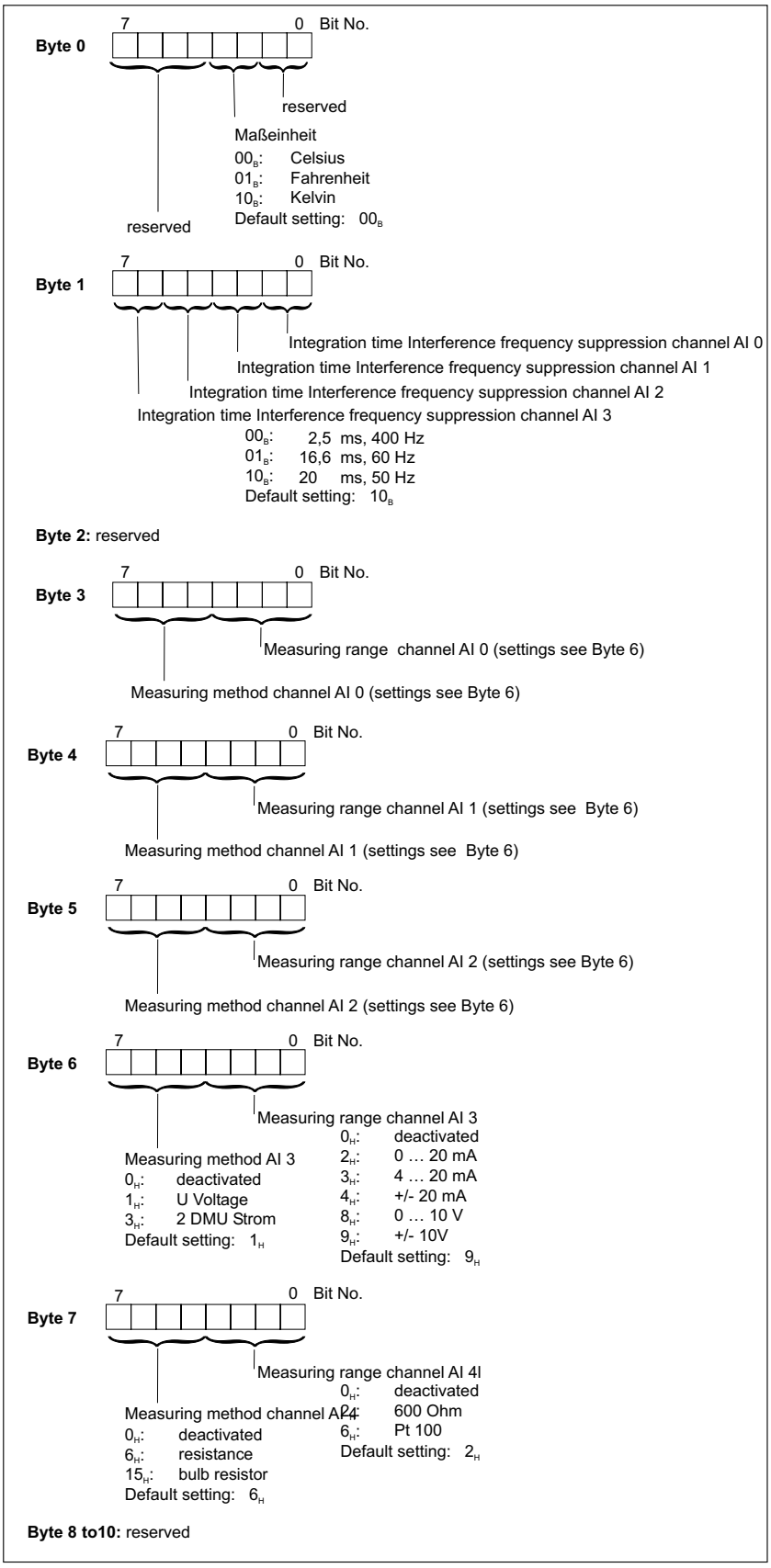
| Parameters | Value range | Default setting | Efficiency range |
|---|---|---|---|
| Integration time (ms)<br>Interference suppression (Hz)<br>(channel 0 to 3) | 2,5/16,6/20<br>400/60/50 | 20<br>50 | Channel<br>Channel |
| Measurement range<br>(channel 0 to 3) | disabled /<br>+/- 20 mA/<br>0 ... 20 mA/<br>4 ... 20 mA/<br>+/- 10 V/<br>0 ... 10 V | +/- 10 V | Channel |
| Type of measurement<br>(channel 0 to 3) | disabled /<br>U voltage /<br>2DMU current | U voltage | Channel |
| Unit of measurement<br>(channel 4) | Centigrade/Fahrenheit/<br>Kelvin | Centigrade | Channel |
| measurement range<br>(Pt 100 input; channel 4) | disabled /<br>Pt 100/600 Ω | 600 Ω | Channel |
| Type of measurement<br>(Pt 100 input; channel 4) | disabled /<br>resistance/<br>thermal resistance | Resistance | Channel |

## Parameters of Standard AO

The following table gives you an overview of the parameters for standard analog outputs (see also Chapter 4.3 in the Reference Manual *Module Data*).

Table 7-4 Parameters of Standard AO

| Parameters | Value range | Default setting | Efficiency range |
|---|---|---|---|
| Output range<br>(channel 0 to 1) | disabled /<br>+/- 20 mA/<br>0 ... 20 mA/<br>4 ... 20 mA/<br>+/- 10 V/<br>0 ... 10 V | +/- 10 V | Channel |
| Type of output<br>(channel 0 to 1) | disabled /<br>U voltage /<br>2DMU current | U voltage | Channel |


Byte 0
7
0
Bit No.
reserved
Maßeinheit
00B: Celsius
01B: Fahrenheit
10B: Kelvin
Default setting: 00B
reserved
Byte 1
7
0
Bit No.
Integration time Interference frequency suppression channel AI 0
Integration time Interference frequency suppression channel AI 1
Integration time Interference frequency suppression channel AI 2
Integration time Interference frequency suppression channel AI 3
00B: 2,5 ms, 400 Hz
01B: 16,6 ms, 60 Hz
10B: 20 ms, 50 Hz
Default setting: 10B
Byte 2: reserved
Byte 3
7
0
Bit No.
Measuring range channel AI 0 (settings see Byte 6)
Measuring method channel AI 0 (settings see Byte 6)
Byte 4
7
0
Bit No.
Measuring range channel AI 1 (settings see Byte 6)
Measuring method channel AI 1 (settings see Byte 6)
Byte 5
7
0
Bit No.
Measuring range channel AI 2 (settings see Byte 6)
Measuring method channel AI 2 (settings see Byte 6)
Byte 6
7
0
Bit No.
Measuring range channel AI 3
0H: deactivated
2H: 0 … 20 mA
3H: 4 … 20 mA
4H: +/- 20 mA
8H: 0 … 10 V
9H: +/- 10V
Default setting: 9H
Measuring method AI 3
0H: deactivated
1H: U Voltage
3H: 2 DMU Strom
Default setting: 1H
Byte 7
7
0
Bit No.
Measuring range channel AI 4I
0H: deactivated
2H: 600 Ohm
6H: Pt 100
Default setting: 2H
Measuring method channel AI 4
0H: deactivated
6H: resistance
15H: bulb resistor
Default setting: 6H
Byte 8 to10: reserved

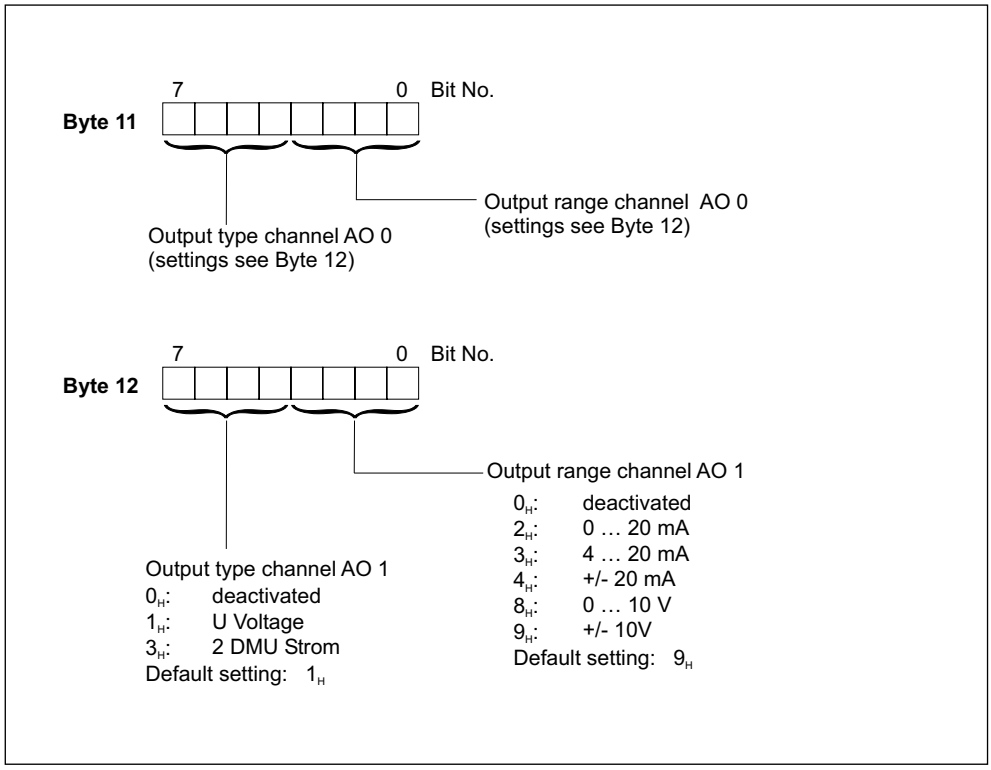



Figure 7-11 Structure of Data Record 1 for Standard AI/AO (length is 13 bytes)

## Parameters for Technological Functions

The parameters for the respective function are found in the Manual *Technological Functions*.

## 7.4 Interrupts

### Interrupt Inputs

All on-board digital inputs of CPUs 31xC can be used as interrupt inputs.

You can specify interrupt behavior for each individual input in your parameter declaration. Options are:

- no interrupt
- interrupt triggered at the positive edge
- interrupt triggered at the negative edge
- interrupt triggered at the positive and negative edge

**Note**

If the rate of incoming interrupts exceeds the handling capacity of OB40, every channel maintains one event. Further events (interrupts) are lost without diagnostics and explicit message.

### Start information for OB40

The table below shows the relevant temporary variables (TEMP) of OB40 for the interrupt inputs of CPUs 31xC. A description of the process interrupt OB40 is found in the Reference Manual *System and Standard Functions*.

Table 7-5 Start information for OB40, relating to the interrupt inputs of the integrated I/O

| Byte | Variable | Data Type | | Description |
|---|---|---|---|---|
| 6/7 | OB40_MDL_ADDR | WORD | B#16#7C | Address of the interrupt-triggering module (here: Default addresses of the digital inputs) |
| 8 on | OB40_POINT_ADDR | DWORD | see the figure below | Signaling of the interrupt triggering integrated inputs |

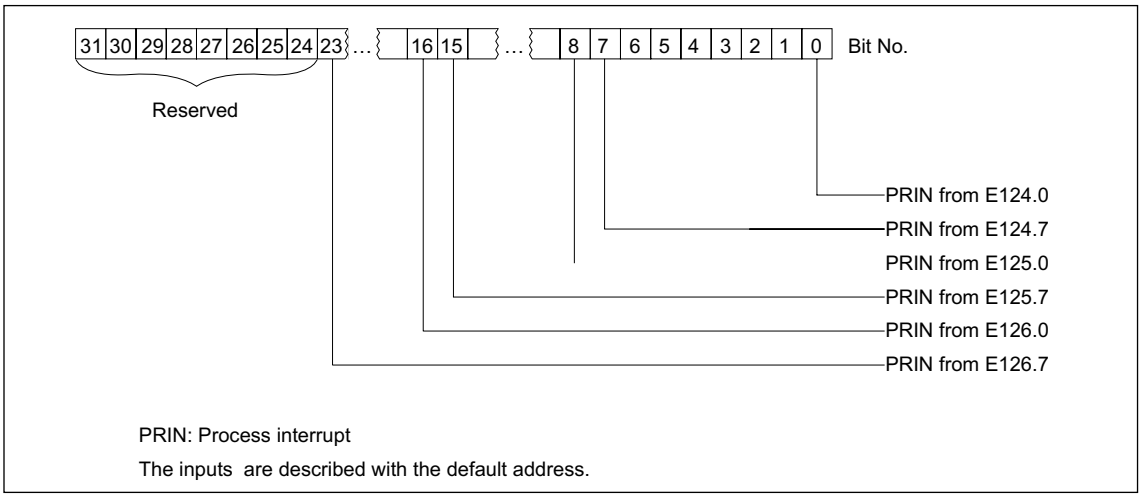



Figure 7-12 Display of the Status of the Interrupt Inputs of CPU 31xC

## 7.5 Diagnostics

### Standard I/O

Diagnostics is not available for integrated I/O used as standard I/O (see also the Reference Manual *Module Data*).

### Technological Functions

Diagnostic options for the respective technological function are found in the Manual *Technological Functions*.

## 7.6 Digital Inputs of CPUs 31xC

### Introduction

This chapter contains the technical data for the digital inputs of CPUs 31xC.

The table includes the following CPUs:

- under CPU 313C-2, the CPU 313C-2 DP and CPU 313C-2 PtP
- under CPU 314C-2, the CPU 314C-2 DP and CPU 314C-2 PtP

### Technical Data

| Technical Data | | | | |
|---|---|---|---|---|
| **Module Specific Data** | **CPU 312C** | **CPU 313C** | **CPU 313C-2** | **CPU 314C-2** |
| Number of inputs | 10 | 24 | 16 | 24 |
| • inputs usable for technological functions | 8 | 12 | 12 | 16 |
| Cable length (for standard DI / Technological Functions) | | | | |
| • Unshielded | max. 600 m / no | | | |
| • Shielded | max. 1,000 m / max. 100 m | | | |
| **Voltage, currents, potentials** | **CPU 312C** | **CPU 313C** | **CPU 313C-2** | **CPU 314C-2** |
| Rated load current L+ | 24V DC | | | |
| • Polarity reversal protection | Yes | | | |
| Number of inputs that can be triggered simultaneously | | | | |
| • horizontal assembly | | | | |
| up to 40C | 10 | 24 | 16 | 24 |
| up to 60C | 5 | 12 | 8 | 12 |
| • vertical assembly | | | | |
| up to 40C | 5 | 12 | 8 | 12 |

| **Technical Data** | | | | |
|---|---|---|---|---|
| Galvanic isolation | | | | |
| • between channels and backplane bus | Yes | | | |
| • between the channels | No | | | |
| Permissible potential difference | | | | |
| • between different circuits | DC 75 V / AC 60 V | | | |
| Insulation tested at | 500V DC | | | |
| Current consumption | | | | |
| • from power voltage L+ (no-load) | – | max. 70 mA | max. 70 mA | max. 70 mA |
| **Status, Interrupts, Diagnostics** | **CPU 312C** | **CPU 313C** | **CPU 313C-2** | **CPU 314C-2** |
| Status display | 1 green LED per channel | | | |
| Interrupts | • yes, if the corresponding channel is configured as interrupt input<br>• when using the technological function refer to the Manual *Technological Functions* | | | |
| Diagnostic functions | • no diagnostics when operated as standard I/O<br>• when using the technological function refer to the Manual *Technological Functions* | | | |
| **Data for the selection of an encoder for standard DI** | **CPU 312C** | **CPU 313C** | **CPU 313C-2** | **CPU 314C-2** |
| Input voltage | | | | |
| • Rated value | 24V DC | | | |
| • for “1” signal | 15 V to 30 V | | | |
| • for “0” signal | -3 V to 5 V | | | |
| Input current | | | | |
| • for “1” signal | normally 9 mA | | | |
| Delay of the standard inputs | | | | |
| • configurable | yes (0.1 / 0.5 / 3 / 15 ms) | | | |
| • Rated value | 3 ms | | | |
| Input delay when using technological functions | 50 µs | 16 µs | 16 µs | 8 µs |
| Input characteristic | to IEC 1131, Type 1 | | | |
| Connection of 2 wire BEROs | Possible | | | |
| • permissible quiescent current | max. 1.5 mA | | | |

## 7.7 Digital outputs of CPUs 31xC

### Introduction

This chapter contains the technical data for the digital outputs of CPUs 31xC.

The table includes the following CPUs:

- under CPU 313C-2, the CPU 313C-2 DP and CPU 313C-2 PtP
- under CPU 314C-2, the CPU 314C-2 DP and CPU 314C-2 PtP

### Fast Digital Outputs

Technological functions utilize the fast digital outputs. These outputs must only be connected to resistive loads.

### Technical Data

| Technical Data | | | | |
|---|---|---|---|---|
| **Module Specific Data** | **CPU 312C** | **CPU 313C** | **CPU 313C-2** | **CPU 314C-2** |
| Number of outputs | 6 | 16 | 16 | 16 |
| • of those are fast outputs | 2 | 4 | 4 | 4 |
| Cable length | | | | |
| • Unshielded | max. 600 m | | | |
| • Shielded | max. 1000 m | | | |
| **Voltage, currents, potentials** | **CPU 312C** | **CPU 313C** | **CPU 313C-2** | **CPU 314C-2** |
| Rated load current L+ | 24V DC | | | |
| • Polarity reversal protection | Yes | | | |
| Total current of outputs (per group) | | | | |
| • horizontal assembly | | | | |
| up to 40C | max. 2.0 A | max. 3.0 A | max. 3.0 A | max. 3.0 A |
| up to 60C | max. 1.5 A | max. 2.0 A | max. 2.0 A | max. 2.0 A |
| • vertical assembly | | | | |
| up to 40C | max. 1.5 A | max. 2.0 A | max. 2.0 A | max. 2.0 A |
| Galvanic isolation | | | | |
| • between channels and backplane bus | Yes | | | |
| • between the channels | No | Yes | Yes | Yes |
| in groups of | – | 8 | 8 | 8 |
| Permissible potential difference | | | | |
| • between different circuits | DC 75 V / AC 60 V | | | |
| Insulation tested at | 500V DC | | | |
| Current consumption | | | | |
| • from load voltage L+ | max. 50 mA | max. 100 mA | max. 100 mA | max. 100 mA |

| Technical Data | | | | |
|---|---|---|---|---|
| **Status, Interrupts, Diagnostics** | **CPU 312C** | **CPU 313C** | **CPU 313C-2** | **CPU 314C-2** |
| Status display | 1 green LED per channel | | | |
| Interrupts | • no interrupts when operated as standard I/O<br>• when using the technological function refer to the Manual *Technological Functions* | | | |
| Diagnostic functions | • no diagnostics when operated as standard I/O<br>• when using the technological function refer to the Manual *Technological Functions* | | | |
| **Data for the selection of an actuator for standard DI** | **CPU 312C** | **CPU 313C** | **CPU 313C-2** | **CPU 314C-2** |
| Output voltage | | | | |
| • for “1” signal | min. L+ (-0.8 V) | | | |
| Output current | | | | |
| • for “1” signal | | | | |
| Rated value | 0.5 A | | | |
| Permissible range | 5 mA to 0.6 A | | | |
| • for “0” signal (residual current) | max. 0.5 mA | | | |
| Load impedance range | 48 Ω to 4 kΩ | | | |
| Lamp load | max. 5 W | | | |
| Parallel connection of 2 outputs | | | | |
| • for redundant load control | Possible | | | |
| • for performance increase | Not possible | | | |
| Triggering of a digital input | Possible | | | |
| Switching frequency | | | | |
| • for resistive load | max. 100 Hz | | | |
| • for inductive load to IEC947 -5 , DC | max. 0.5 Hz | | | |
| • for lamp load | max. 100 Hz | | | |
| • fast outputs with resistive load | max. 2.5 kHz | | | |
| Inductive breaking voltage limited internally to | normally (L+) - 48 V | | | |
| Short-circuit protected output | yes, electronic | | | |
| • Response threshold | normally 1 A | | | |

## 7.8 Analog Inputs of CPUs 31xC

### Introduction

This chapter contains the technical data for the analog outputs of CPUs 31xC.

The table includes the following CPUs:

- under CPU 314C-2, the CPU 314C-2 DP and CPU 314C-2 PtP

### Technical Data

| Technical Data | | |
|---|---|---|
| **Module Specific Data** | **CPU 313C** | **CPU 314C-2** |
| Number of inputs | 4 channels for current/voltage input<br>1 channel for resistance input | |
| Cable length | | |
| • Shielded | max. 100 m (109 yd.) | |
| **Voltage, currents, potentials** | **CPU 313C** | **CPU 314C-2** |
| Resistance Input | | |
| • Idle voltage | normally V 2.5 | |
| • Measurement current | normally 1.8 mA to 3.3 mA | |
| Galvanic isolation | | |
| • between channels and backplane bus | Yes | |
| • between the channels | No | |
| Permissible potential difference | | |
| • between inputs and $M_{ANA}$ ($U_{CM}$) | 1.0V DC | |
| • between $M_{ANA}$ and $M_{internally}$ ($U_{ISO}$) | DC 75 V / AC 60 V | |
| Insulation tested at | 600V DC | |
| **Analog Value Generation** | **CPU 313C** | **CPU 314C-2** |
| Measuring principle | Momentary value encoding (successive approximation) | |
| Integration time/Conversion time/Resolution (per channel) | | |
| • configurable | Yes | |
| • Integration time in ms | 2,5 / 16,6 / 20 | |
| • Permissible input frequency | max. 400 Hz | |
| • Resolution (including overdrive) | 11 bits + sign bit | |
| • Suppression of interference frequency f1 | 400 / 60 / 50 Hz | |
| Time constant of the input filter | 0.38 ms | |
| Basic execution time | 1 ms | |

| Technical Data | | |
|---|---|---|
| **Interference Suppression, Error Limits** | **CPU 313C** | **CPU 314C-2** |
| Interference voltage suppression for f = n x (f1 ± 1 %), (f1 = interference frequency), n = 1, 2 | | |
| • Common mode interference ($_{UCM}$<1.0 V) | > 40 dB | |
| • Feedback interference (peak value of the interference < rated value of the input range) | > 30 dB | |
| Crosstalk between the inputs | > 60 dB | |
| Operational error limits (throughout temperature range, relative to input range) | | |
| • Voltage/Current | < 1 % | |
| • Resistance | < 5% | |
| Basic error limits (operational limit at 25C, relative to input range) | | |
| • Voltage/Current | < 0.7 % | |
| • Resistance | < 3% | |
| Temperature error (related to input range) | ± 0.006%/K | |
| Linearity error (related to input range) | ± 0.06% | |
| Accuracy of reproducibility (in transient state at 25 °C, relative to input range) | ± 0.06% | |
| **Status, Interrupts, Diagnostics** | **CPU 313C** | **CPU 314C-2** |
| Interrupts | • no interrupts when operated as standard I/O | |
| Diagnostic functions | • no diagnostics when operated as standard I/O<br>• when using the technological function refer to the Manual *Technological Functions* | |
| **Encoder Selection Data** | **CPU 313C** | **CPU 314C-2** |
| Input ranges (rated value)/input resistance | | |
| • Voltage | ± 10 V/100 kΩ<br>0 V to 10 V/100 kΩ | |
| • Current | ± 20 mA/50 Ω<br>0 mA to 20 mA/50 Ω<br>4 mA to 20 mA/50 Ω | |
| • Resistance | 0 Ω to 600 Ω/10 MΩ | |
| • Resistive thermometer | Pt 100/10 MΩ | |
| permissible input voltage (destruction limit) | | |
| • for voltage inputs | max. " 50 V continuous; | |
| • for current inputs | max. " 2.5 V continuous; | |
| permissible input current (destruction limit) | | |
| • for voltage inputs | max. " 0.5 mA continuous; | |
| • for current inputs | max. " 50 mA continuous; | |
| Connection of Signal Generators | | |
| • for voltage measurement | Possible | |

| Technical Data | |
|---|---|
| • for current measurement | |
| as 2-wire measuring transducer | possible, with external power supply |
| as 4-wire measuring transducer | Possible |
| • for measuring resistance | |
| with 2-wire connection | possible, without cable resistance compensation |
| with 3-wire connection | Not possible |
| with 4-wire connection | Not possible |
| Linearization of the Characteristic Curve | by software |
| • for resistive thermometer | Pt 100 |
| Temperature compensation | No |
| Technical unit for temperature measurement | Degrees Centigrade/Fahrenheit/Kelvin |

## 7.9 Analog outputs of CPUs 31xC

### Introduction

This chapter contains the technical data for the digital outputs of CPUs 31xC.

The table includes the following CPUs:

- under CPU 314C-2, the CPU 314C-2 DP and CPU 314C-2 PtP

### Technical Data

| Technical Data | | |
|---|---|---|
| **Module Specific Data** | **CPU 313C** | **CPU 314C-2** |
| Number of outputs | 2 | |
| Cable length | | |
| • Shielded | max. 200 m | |
| **Voltage, currents, potentials** | **CPU 313C** | **CPU 314C-2** |
| Rated load current L+ | 24V DC | |
| • Polarity reversal protection | Yes | |
| Galvanic isolation | | |
| • between channels and backplane bus | Yes | |
| • between the channels | No | |
| Permissible potential difference | | |
| • between outputs and $M_{ANA}$ ($U_{CM}$) | 1.0V DC | |
| • between $M_{ANA}$ and $M_{internally}$ ($U_{ISO}$) | DC 75 V, AC 60 V | |
| Insulation tested at | 600V DC | |

| Technical Data | | |
|---|---|---|
| **Analog Value Generation** | **CPU 313C** | **CPU 314C-2** |
| Resolution (including overdrive) | 11 bits + sign bit | |
| Conversion time (per channel) | 1 ms | |
| Settling time | | |
| • for resistive load | 0.6 ms | |
| • for capacitive load | 1.0 ms | |
| • for inductive load | 0.5 ms | |
| **Interference Suppression, Error Limits** | **CPU 313C** | **CPU 314C-2** |
| Crosstalk between the outputs | > 60 dB | |
| Operational error limits (throughout temperature range, relative to output range) | | |
| • Voltage/Current | ± 1 % | |
| Basic error limit (operational limit at 25C, relative to output range) | | |
| • Voltage/Current | ± 0.7% | |
| Temperature error (relative to output range) | ± 0.01%/K | |
| Linearity error (relative to output range) | ± 0.15% | |
| Accuracy of reproducibility (in transient state at 25C, relative to output range) | ± 0.06% | |
| Output ripple; Bandwidth 0 to 50 kHz (relative to output range) | ± 0.1% | |
| **Status, Interrupts, Diagnostics** | **CPU 313C** | **CPU 314C-2** |
| Interrupts | • no interrupts when operated as standard I/O<br>• when using the technological function refer to the Manual *Technological Functions* | |
| Diagnostic functions | • no diagnostics when operated as standard I/O<br>• when using the technological function refer to the Manual *Technological Functions* | |
| **Actuator Selection Data** | **CPU 313C** | **CPU 314C-2** |
| Output range (rated values) | | |
| • Voltage | ± 10 V<br>0 V to 10 V | |
| • Current | ± 20 mA/<br>0 mA to 20 mA/<br>4 mA to 20 mA/ | |
| Load resistance (in the rated range of the output) | | |
| • with voltage outputs | min. 1 kΩ | |
| capacitive load | max. 0.1 µF | |
| • with current outputs | max. 300 Ω | |
| inductive load | 0.1 mH | |
| Voltage output | | |

| **Technical Data** | |
|---|---|
| • Short-circuit protection | Yes |
| • Short-circuit current | normally 55 mA |
| Current output | |
| • Idle voltage | normally 17 V |
| Destruction limit for externally applied voltages/currents | |
| • Output voltage to $M_{ANA}$ | max. 16 V continuous; |
| • Current | max. " 50 mA continuous; |
| Connection of actuators | |
| • for voltage output | |
| 2-wire connection | possible, without cable resistance compensation |
| 4-wire connection (measuring line) | Not possible |
| • for current output | |
| 2-wire connection | Possible |

# Migration from CPU 31x to CPU 31xC

# 8

You might meet the following problems if you transfer your existing user program for the CPU 31x to a CPU 31xC:

## SFC with asynchronous operation

Some of the SFCs operating asynchronously on CPUs 31x were always or under specific conditions processed after the first call ("quasi-synchronous").

These SFC run on CPUs 31xC really asynchronously. Asynchronous processing might cover multiple OB1 cycles, with the result that a wait loop can develop into an endless loop within the OB.

Affected are:

- SFC 56 "WR_DPARM"; SFC 57 "PARM_MOD"

  These SFC always operate in "quasi-synchronous" mode on central CPUs 31x. On central CPUs 31xC and distributed CPUs it operates in asynchronous mode.

- SFC 13 "DPNRM_DG"

  This SFC always operates in "quasi-synchronous" mode when called in OB82. On CPUs 31xC it always operates asynchronously.

## Restriction of SFC functions

SFC 20 "BLKMOV"

This SFC could previously called to copy data from a runtime irrelevant DB.

SFC20 does not have this functionality anymore. SFC83 "READ_DBL" has now replaced its functionality.

## SFC not available anymore

SFC 54 "RD_DPARM"

This SFC is not available anymore. It was replaced by the asynchronously operating SFC102 "RD_DPARA".

### SFCs possibly delivering other results

Ignore the following points if you use only logical addressing in your user program.

If you use address conversions in your user program (SFC 5 "GADR_LGC", SFC 49 "LGC_GADR"), you have to check the assigment of slots and the logical start address for DP slaves.

- The diagnostic address of the DP slave is now always assigned slot 0.
- DP slave is integrated in STEP 7:

The interface module (slot 2) has its own address (e. g. CPU 31x-2DP as I slave).

### Changing diagnostic addresses of DP slaves

Note that the diagnostic addresses for the slaves have to be assigned

again when using CPU 31xC-2DP as master since 2 diagnostic addresses are required per slave in part.

- The virtual slot 0 has its own address.

  The module status data of this slot (read out with SFC 51 "RDSYSST") contains the identifiers, which concerns the entire slave/the entire station, e. g. the ID station is interrupted.
- In addition, slot 2 also has its own address for the modules integrated in STEP 7 (e. g. CPU as I Slave). For example, with CPU 31x-2DP as I slave the mode change in the diagnostic interrupt OB 82 of the master is sent via this address.

---

**Note**
Reading out diagnosis with SFC 13 "DPNRM_DG":

The diagnostic address originally assigned also continues to function. Internally STEP 7 assigns this address slot 0.

---

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

### Using consistent data areas in the Process Image for DP slaves

For this, read the section on consitent data in chapter Addressing of the installation manual.

# Glossary

# 9

### Accumulator

The --> CPU uses the accumulator registers as intermediate memory for load, transfer, comparison, calculation and conversion operations.

### Address

The address represents the ID for a specific operand or operand range. Example: Input I 12.1; Memory bit Word MW25; Data block DB3.

### Analog module

Analog modules convert process values (e.g. temperature) into digital values, so that they can be processed by the central processing unit, or convert digital values into analog manipulated variables.

### Automation system

An automation system in the context of SIMATIC S7 --> is a programmable logic controller.

### Backplane Bus

The serial backplane data bus supplies the power required by the modules. It is also used by the modules for communication. The connection between the modules is established by bus connectors.

### Backup memory

This memory is used to backup memory areas of --> CPUs not equipped with a backup battery. A configurable number of timers, counters, memories and data bytes (retentive timers, counters, memories and data bytes) is backed up.

### Bus

A bus is a communication medium connecting several nodes. Data transmission can be serial or parallel across electrical conductors or optical waveguides.

### Bus segment

A bus segment is a self-contained section of a serial bus system. Bus segments are interconnected using repeaters.

### Chassis ground

Chassis ground is the totality of all the interconnected inactive parts of a piece of equipment on which a hazardous touch voltage cannot build up even in the event of a fault.

### Clock memories

Memories that can be used for clocking purposes in the user program (1 memory byte).

**Note**

Note in the case of S7 300 CPUs that the clock memory byte is not overwritten in the user program.

### Code block

A SIMATIC S7 code block contains part of the STEP 7 user program. (In contrast: a --> data blocks (DB) only contain data.)

### Communication processor

Communication processors are modules for point-to-point and bus communication.

### Compress

The PG's online function “Compress” is used to align all valid blocks contiguously in the RAM of the CPU at the start of the user memory. This eliminates all gaps which arose when blocks were deleted or modified.

### Configuration

Assignment of modules to racks/slots and (e.g. for signal modules) addresses.

### Consistent data

Data whose contents are related and which should not be separated are known as consistent data.

For example, the values of analog modules must always be handled consistently, that is the value of an analog module must not be corrupted by reading it out at two different times.

### Counter

Counters are part of CPU --> system memory. The content of "Counter cells" can be modified by **STEP 7** instructions (e.g. up/down count).

### CP

--> Communication Processor

### CPU

Central Processing Unit of an S7 PLC, consisting of the control and arithmetic unit, memory, operating system and a PG interface.

### Cycle Time

The term cycle time describes the time required by a --> CPU for one --> user program execution.

### Data block

Data blocks (DB) are data areas in the user program which contain user data. Global data blocks can be accessed by all code blocks while instance data blocks are assigned to a specific FB call.

### Data, static

Static data is data which can only be used within a function block. The data is saved in an instance data block belonging to the function block. The data stored in the instance data block is retained until the next function block call.

### Data, temporary

Temporary data is local data of a block that is stored in the L stack during block execution and no longer available after execution.

### Delay Interrupt

--> Interrupt, Delay

### Diagnostic buffer

The diagnostic buffer is a buffered memory area in the CPU in which diagnostic events are stored in the order of their occurrence.

### Diagnostic Interrupt

Modules capable of diagnostic operations report detected system error events to the --> CPU, using diagnostic interrupts.

### Diagnostics

--> System Diagnostics

### DP master

A --> master which operates in accordance with EN 50170, Part 3 is referred to as a DP master.

### DP Slave

A --> slave operated on PROFIBUS with PROFIBUS DP protocol and in accordance with EN 50170, Part 3 is referred to as DP slave.

### Equipotential bonding

Electrical connection (equipotential bonding conductor) which gives the bodies of electrical equipment and external conducting bodies the same or approximately the same potential, in order to prevent disturbing or dangerous voltages from being generated between these bodies.

### Error display

The error display is one of the possible responses of the operating system to a --> runtime error. The other possible responses are: --> error response in the user program, CPU STOP.

### Error handling via OB

When the operating system detects a specific error (e.g. access error with STEP 7), it calls a dedicated organization block (Error OB) that determines subsequent CPU response.

### Error response

Response to a --> runtime error. The operating system can respond in the following ways: transition of the PLC to STOP mode, call of an organization block in which the user can program an error response or display.

### External power supply

Power supply for the signal and function modules and the I/O connected to them.

### FB

--> Function Block

### FC

--> Function

### Flash EPROM

FEPROMs are the same as electrically erasable EEPROMS in that they can retain data in the event of a power failure, but they can be erased much more quickly (FEPROM = Flash Erasable Programmable Read Only Memory). They are used on --> Memory Cards.

### Force

The "Force" function overwrites a variable (e.g. memory bit, output) with a value defined by the S7 user. At the same time, this variable is write protected, thus preventing modification by any other operation (including from the STEP 7 user program). The value is retained after the programming device is disconnected. The write protection can only be cleared with the "Unforce" function. The value specified in the user program is then written back to the variable. For example, during commissioning you can use the "Force" function to set specific outputs to "ON" state for an indefinite time, even if the user program is not logically linked (e.g. inputs are not wired).

### Function

According to IEC 1131-3, a function is a --> code block that contains no --> static data. A function allows parameters to be passed in the user program. Functions are therefore suitable for programming complex functions, e.g. calculations, which are repeated frequently.

### Function block

According to IEC 1131-3, a function block is a --> code block that contains --> static data. An FB allows parameters to be passed in the user program. Function blocks are therefore suitable for programming complex functions, e.g. closed-loop controls, mode selections, which are repeated frequently.

### Functional grounding

Grounding which has the sole purpose of safeguarding the intended function of the electrical equipment. Functional grounding short-circuits interference voltage which would otherwise have an impermissible impact on the equipment.

### Galvanically isolated

The reference potential of the control and on-load power circuits for isolated I/Os is galvanically isolated; e.g. by optocouplers, relay contact or transformer. I/O circuits can be connected to a common potential.

### GD circuit

A GD circuit consists of a number of CPUs exchanging data by means of global data communication and which are used as follows:

One CPU broadcasts a GD packet to the other CPUs.

One CPU sends and receives a GD packet from another CPU.

A GD circuit is identified by a GD circuit number.

### GD Element

A GD element is generated by assigning shared --> global data. It is identified by a unique global data ID in the global data table.

### GD packet

A GD packet can consist of one or multiple --> GD elements transferred in a single message frame.

### Global data communication

Global data communication is a procedure used for --> global data exchange between CPUs (no CFBs).

### Ground

The conducting earth whose electrical potential can be set equal to zero at any point.

In the vicinity of grounding electrodes, the earth can have a potential different to zero. The term “reference ground” is frequently used to describe these circumstances.

### Ground (to)

To ground means to connect an electrically conducting component to the grounding electrode (one or more conducting components which have a very good contact with the earth) across a grounding system.

### GSD file (device master file)

The device master file (GSD file) stores all slave specific properties. The GSD file format is specified in EN 50170,Volume 2, PROFIBUS.

### Instance data block

A DB is automatically generated and assigned to every function block call in the STEP 7 user program. The values of the input, output and in/out parameters are stored in the instance data block, together with local block data.

### Interface, multipoint

--> MPI

### Interrupt

The CPU's --> operating system knows 10 different priority classes for controlling user program execution. e.g. process interrupts. When an interrupt is triggered, the operating system automatically calls an assigned organization block in which the user can program the desired response (for example in an FB).

### Interrupt, delay

The delay interrupt belongs to one of the priority classes when processing programs in SIMATIC S7. It is started on expiration of a time generated in the user program. A corresponding organization block is then executed.

### Interrupt, Diagnostics

--> Diagnostic Interrupt

### Interrupt, Process

--> Process Interrupt

### Interrupt, time-of-day

The time-of-day interrupt belongs to one of the priority classes when processing programs in SIMATIC S7. It is generated depending on a specific date (or daily) and time-of-day (e.g. 9:50 or hourly, or every minute). A corresponding organization block is then executed.

### Interrupt, watchdog

A watchdog interrupt is generated periodically by the CPU in configurable time intervals. A corresponding --> organization block is then executed.

### Load memory

The load memory is part of the central processing unit. It contains objects generated by the programming device. It is implemented either as a plug-in memory card or a permanently integrated memory.

### Local data

--> Data, temporary

### Main memory

Working memory is a RAM memory in the --> CPU accessed by the processor during user program execution.

### Master

Masters in possession of the --> Token can send/request data to/from other nodes (= active node).

### Memory bits

Memory bits are part of the CPU's --> system memory. They store the intermediate results of calculations. They can be accessed in bit, byte, word or doubleword units.

### Memory card

Memory cards are memory media in smart card format for CPUs and CPs. They are implemented as --> RAM or --> FEPROM.

### Micro Memory Card (MMC)

Micro Memory Cards are memory media for CPUs and CPs. Its smaller dimensions form the only difference compared to the --> Memory Card.

### Module parameters

Module parameters are values which can be used to control the response of the module. A distinction is made between static and dynamic module parameters.

### MPI

This interface is capable of multipoint communication (MPI). It forms part of the SIMATIC S7 PG interface. it also offers optional multiple node operation (PGs, text displays, operator panels) on one or several PLCs. Each node is identified by a unique address (MPI address).

### MPI address

--> MPI

### Nesting depth

One block can be called from another by means of a block call. Nesting depth is defined as the number of simultaneously called --> code blocks.

### Non-isolated

The reference potential of the control and on-load power circuits for non-isolated I/Os is electrically interconnected.

### OB

--> Organization Blocks

### OB priority

The CPU --> operating system distinguishes between different priority classes, e.g. cyclic program execution, program execution controlled by process interrupt. Each priority class is assigned --> organization blocks (OBs) in which the S7 user can program a response. The OBs have different standard priorities which determine the order in which they are executed or interrupted in the event that they are activated simultaneously.

### Operating mode

SIMATIC S7 PLC operating modes are: STOP, --> START-UP, RUN.

### Operating system of the CPU

The operating system of the CPU organizes all functions and processes of the CPU which are not associated with a special control task.

### Organization Blocks

Organization blocks (OBs) represent the interface between the operating system of the CPU and the user program. The processing sequence of the user program is defined in the organization blocks.

### Parameters

1. Variable of a **STEP 7** code block
2. Variable for declaring module response (one or several per module). All modules are delivered with a suitable factory setting which can be changed in the configuration with**STEP 7**.
There are --> static parameters and --> dynamic parameters

### Parameters, dynamic

Unlike static parameters, dynamic parameters of modules can be changed during operation by calling an SFC in the user program, for example limit values of an analog signal input module.

### Parameters, static

Unlike dynamic parameters, static parameters of modules cannot be changed by the user program, but rather by changing the configuration in **STEP 7**, for example the input delay on a digital signal input module.

### PG

--> Programming Device

### PLC

--> Programmable controller

### 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. If a priority class includes several OBs, they do not interrupt each other, but are executed sequentially.

### Process Image

The process image is part of CPU --> system memory. At the start of cyclic program execution, the status of the signal module inputs is written to the input process image. At the end of cyclic program execution, the signal status of the output process image is transferred to the output modules.

### 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 is then processed, according to interrupt priority.

### Product version

The product version differentiates between products which have the same order number. The product version is increased with each upwardly compatible function extension, production-related modification (use of new components) or bug-fix.

### PROFIBUS-DP

The PLC distributes controls for digital, analog and intelligent modules as well as a wide range of field devices to EN 50170, part 3, for example, drives or valve blocks, to processes at external locations - even across distances exceeding 23 km.

The modules and field devices are connected to the programmable controller via the PROFIBUS DP fieldbus and addressed in the same way as centralized I/Os.

### Programmable controller

Programmable controllers (PLCs) are electronic controllers whose function is saved as a program in the control unit. The configuration and wiring of the unit are therefore independent of the function of the control system. The PLC has a computer structure; it consists of the --> CPU (Central Processing Unit) with memories, I/O modules and internal bus system. The I/Os and the programming language are oriented to control engineering needs.

### Programming device

Programming devices are essentially personal computers which are compact, portable and suitable for industrial applications. They are equipped with special hardware and software for SIMATIC PLCs.

### RAM

RAM (Random Access Memory) is a semiconductor read/write memory.

### Reduction factor

The reduction rate determines the send/receive frequency for --> GD packets on the basis of the CPU cycle.

### Reference ground

--> Ground

### Reference potential

Potential with reference to which the voltages of participating circuits are observed and/or measured.

### Restart

When a central processing unit is started up (e.g. by switching the mode selector from STOP to RUN or by switching the power on), organization block OB 100 (complete restart) is executed before cyclic program execution commences (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 OB 1.

### Retentivity

A memory area is retentive if its contents are retained even after a power failure and a change from STOP to RUN. The volatile area of memory bits, timers and counters is reset after a power failure and transition from STOP to RUN mode.

The following can be made retentive:

- Bit memories
- S7 timers
- S7 counters
- Data areas (only with Memory Card or integrated EPROM)

### Runtime error

These are PLC errors occurring during user program execution (that is, not in the process itself).

### Segment

--> Bus Segment

### SFB

--> System function block

### SFC

--> System function

### Shared data

Shared data can be addressed from any --> code block (FC, FB, OB). In detail, this refers to memories M, inputs I, outputs Q, timers, counters and data blocks DB. Absolute or symbolic access can be made to shared data.

### Signal module

Signal modules (SM) form the interface between the process and the PLC. There are digital and analog I/O modules (input/output module, digital or analog).

### Slave

A slave can only exchange data on --> Master request.

### startup

RESTART mode is activated on a transition from STOP mode to RUN mode. Can be triggered by the --> mode selector switch or after power on, or by an operator action on the programming device. An S7–300 performs --> a restart.

### STEP 7

Programming language for developing user programs for SIMATIC S7 PLCs.

### Substitute value

Substitute values are configurable values which output modules transmit to the process when the CPU switches to STOP mode. In the event of an input access error, a substitute value can be written to the accumulator instead of the input value which could not be read (SFC 44).

### System diagnostics

System diagnostics is the term used to describe the detection, evaluation and signaling of errors which occur within the programmable controller. Examples of such errors are program errors or module failures. System errors can be displayed with LED indicators or in **STEP 7**.

### System function

A system function (SFC) is an integrated --> Function of the CPU. It can be called at any time in the STEP 7 user program.

### System function block

A System Function Block (SFB) is a --> function block integrated in the CPU operating system. If required, it can be called in the STEP 7 user program.

### System Memory

The system memory (RAM) is integrated on the central processing unit. System memory contains the operand areas (e.g. timers, counters, memory bits) as well as the data areas required internally by the --> operating system (e.g. buffers for communication).

### System state list

The system status list contains data describing the current status of an S7 300.

You can use it to gain an overview of the following at any time:

- The S7 300 configuration
- The current CPU configuration and the configurable signal modules
- Current status and processes in the CPU and configurable signal modules.

### Terminating resistor

The terminating resistance is used to avoid reflections on data links.

### Time of day interrupt

--> Interrupt, Time-of-day

### Timer

--> Timers, Principle

### Timers

Timers are part of CPU --> system memory. The contents of the “timer cells” are updated automatically by the operating system asynchronously to the user program. **STEP 7** instructions are used to define the exact function of the timer cells (for example on-delay) and initiate their execution (e.g. start).

### Token

Access right on bus

### Transmission rate

Rate of data transfer (bps)

### Ungrounded

Having no galvanic connection to ground

### User memory

User memory contains --> code and --> data blocks of the user program. User memory can be integrated in the CPU or stored on plug-in memory cards or memory modules. However, user programs are always executed from --> CPU working memory.

### User program

The SIMATIC system distinguishes between the --> CPU operating system and user programs. The latter are created with --> -->STEP 7 programming software in optional programming languages (LAD and STL) and stored in code blocks. data is stored in data blocks.

### Varistor

Voltage-independent resistor

### Watchdog Interrupt

--> Interrupt, Watchdog

# Index

## D



## E



## F



## G



## H



## I

## S



## T



## U

## W