**JUMO dTRON 304**
**JUMO dTRON 308**
**JUMO dTRON 316**

Compact Controller with program function

**B 70.3041.2.3**
Interface Description
PROFIBUS-DP

10.05/00450483
## Contents

5.2 Setting the slave address ............................................................... 28
5.3 Diagnostic and status messages .................................................. 28
5.4 Acyclic data transmission ............................................................ 28
  5.4.1 Arrangement of the protocol ................................................ 30
  5.4.2 Demo program dTRON3xx.zip ............................................. 33
  5.4.3 Block diagram of the demo program .................................... 37
  5.4.4 Timing of acyclic data ........................................................... 38
  5.4.5 Commands (GSD generator) ................................................ 38
1 Introduction

1.1 Preface

These operating instructions are addressed to equipment manufacturers with appropriate technical training and PC knowledge.

Please read this Interface Description before starting to work with PROFIBUS-DP. Keep the interface description in a place which is accessible to all users at all times. Your comments will help us improve this interface description, where necessary.

Phone: +49 661 6003-0
Fax: +49 661 6003-607
e-mail: mail@jumo.net

Warranty

All necessary settings are described in this manual. If any difficulties should arise during commissioning, you are asked not to carry out any manipulations that are not described in these operating instructions. You could endanger your rights under the instrument warranty. Please contact the nearest subsidiary or the head office in such a case.

Electrostatic charge

When carrying out work inside the instrument, and when returning modules, assemblies or components, the regulations of EN 61340-5-1 and EN 61340-5-2 “Protection of electronic devices from electrostatic phenomena” must be observed. Only use ESD packaging for transport.

Please note that we cannot accept any liability for damage caused by ESD (electrostatic discharge).
1 Introduction

1.2 Typographical conventions

1.2.1 Warning signs

**Danger**

This sign is used when there may be danger to personnel if the instructions are ignored or not followed correctly!

**Caution**

This sign is used when there may be damage to equipment or data if the instructions are ignored or not followed correctly!

**ESD**

This sign is used where special care is required when handling components liable to damage through electrostatic discharge.

1.2.2 Note signs

**Note**

This sign is used when your special attention is drawn to a remark.

**Reference**

This sign refers to further information in other manuals, chapters or sections.

**Footnote**

Footnotes are remarks that refer to specific points in the text. Footnotes consist of two parts: A marker in the text, and the footnote text itself. The markers in the text are arranged as continuous superscript numbers.
1 Introduction

1.2.3 Performing an action

**Action instruction**

This sign indicates that an action to be performed is described. The individual steps are marked by this asterisk, e.g.

- Start the PLC software
- Click on hardware catalog

**Vital text**

This text contains important information, and it is vital that you read it before going any further.

**Command sequence**

File ➔ Save as

Small arrows between the words indicate a sequence of commands which must be performed one after another.
1 Introduction
PROFIBUS-DP description

PROFIBUS-DP is a manufacturer-independent, open fieldbus standard for a wide range of applications in manufacturing, process and building automation. Manufacturer independence and openness are ensured by the international standards IEC 61158 and IEC 61784. Using PROFIBUS-DP, devices from different manufacturers can communicate without any special interface adjustments. PROFIBUS-DP can be employed for both high-speed time-critical data transmission and extensive, complex communications tasks. The PROFIBUS-DP family consists of three variants.

2.1 Profibus types

<table>
<thead>
<tr>
<th>PROFIBUS-FMS</th>
<th>PROFIBUS-DP</th>
<th>PROFIBUS-PA</th>
</tr>
</thead>
<tbody>
<tr>
<td>universal</td>
<td>fast</td>
<td>sector-oriented</td>
</tr>
<tr>
<td>- wide range of applications</td>
<td>- plug and play</td>
<td>- bus supply</td>
</tr>
<tr>
<td>- multi-master communication</td>
<td>- efficient and cost-effective</td>
<td>- intrinsic safety</td>
</tr>
</tbody>
</table>

The PROFIBUS family

PROFIBUS-DP

This PROFIBUS variant, which is optimized for high speed and low connection costs, has been especially designed for communication between automation control systems (PLC) and distributed field devices (typical access time < 10msec). PROFIBUS-DP can be used to replace conventional, parallel signal transmission with 24V or 0/4 — 20mA.

DPV0: cyclic data transfer:
--> is supported by all JUMO instruments (devices).

DPV1: cyclic and acyclic data transfer:
--> is not supported by JUMO instruments (devices).

DPV2: slave-to-slave communication takes place in addition to cyclic and acyclic data transfer:
--> is not supported by JUMO instruments (devices).

PROFIBUS-PA

PROFIBUS-PA has been specifically designed for process engineering. It permits the linking of sensors and actuators to a common bus cable, even in hazardous areas. PROFIBUS-PA enables data communication and energy supply for devices in two-wire technology according to MBP (Manchester Bus Powering) specified in the international IEC 61158-2 standard.

PROFIBUS-FMS

This is the universal solution for communication tasks at cell level (typical access time: approx. 100msec). The powerful FMS services open up a wide range of applications and provide a high degree of flexibility. FMS is also suitable for extensive communication tasks.
2 PROFIBUS-DP description

2.2 RS485 transmission technology

Transmission takes place according to the RS485 standard. It covers all areas in which high transmission speed and simple, cost-effective installation are required. A shielded twisted copper cable with one conductor pair is used.

The bus structure permits addition and removal of stations or step-by-step commissioning of the system without affecting the other stations. Later expansions have no influence on the stations which are already in operation.

Transmission speeds between 9.6kbit/sec and 12Mbit/sec are available. One uniform transmission speed is selected for all devices on the bus when the system is commissioned.

<table>
<thead>
<tr>
<th>Basic characteristics</th>
<th>Network topology</th>
<th>linear bus, active bus termination at both ends, stub cables are only permissible for baud rates &lt;1.5 Mbit/sec.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Medium</td>
<td>shielded twisted-pair cable</td>
<td></td>
</tr>
<tr>
<td>Number of stations</td>
<td>32 stations in each segment without repeater (line amplifier). With repeaters, this can be expanded to 126 (a maximum of 9 repeaters is possible)</td>
<td></td>
</tr>
<tr>
<td>Connector</td>
<td>preferably 9-pin sub-D connector</td>
<td></td>
</tr>
</tbody>
</table>

| Structure | All devices must be connected in a line structure (one after another). Up to 32 stations (master or slaves) can be linked up within one segment. If there are more than 32 stations, repeaters must be used to further increase the number of devices, for instance. |

| Cable length | The maximum cable length depends on the transmission speed. The cable length specified can be extended by using repeaters. It is recommended not to connect more than 3 repeaters in series. |

<table>
<thead>
<tr>
<th>Baud rate (kbit/sec)</th>
<th>9.6</th>
<th>19.2</th>
<th>93.75</th>
<th>187.5</th>
<th>500</th>
<th>1500</th>
<th>12000</th>
</tr>
</thead>
<tbody>
<tr>
<td>Range/segment</td>
<td>1200 m</td>
<td>1200 m</td>
<td>1200 m</td>
<td>1000 m</td>
<td>400 m</td>
<td>200 m</td>
<td>100 m</td>
</tr>
</tbody>
</table>

| Bus termination | The bus is terminated by termination resistors at the start and end of each segment. Both bus terminators must always be powered, to ensure fault-free operation. The termination resistors are located in the Profibus connectors and can be activated by setting the slide switch to ON. |

![Diagram of PROFIBUS-DP termination resistors]
These cable length specifications refer to the cable type A described below:

- Characteristic impedance: 135 — 165 Ω
- Capacitance per unit length: < 30 pf/m
- Loop resistance: 110 Ω/km
- Core dia.: 0.64 mm
- Core cross-section: > 0.34 mm²

It is preferable to use a 9-pin sub-D connector for PROFIBUS networks incorporating RS485 transmission technology. The PIN assignment at the connector and the wiring are shown at the end of this chapter.

PROFIBUS-DP cables and connectors are supplied by several manufacturers. Please refer to the PROFIBUS product catalog (www.profibus.com) for types and addresses of suppliers.

When connecting up the devices, make sure that the data lines are not reversed. It is absolutely essential to use shielded data lines. The braided shield and the screen foil underneath (if present) should be connected to the protective earth on both sides, and with good conductivity. Furthermore, the data lines should be routed separately from all high-voltage cables, as far as this is possible.

As a suitable cable we recommend the following type from Siemens:

**Simatic Net Profibus 6XV1**
**Order No. 830-0AH10**
* (UL) CMX 75 °C ( Shielded) AWG 22 *
2 PROFIBUS-DP description

Data rate

For installation, the use of stub cables must be avoided for data rates above 1.5 Mbit/sec.

For important notes on installation, please refer to the Installation Guidelines PROFIBUS-DP, Order No. 2.111 by the PNO (Profibus Nutzerorganisation).

Address:
Profibus Nutzerorganisation e.V.
Haid- u. Neu-Straße 7
D-76131 Karlsruhe, Germany

Internet: www.profibus.com

Recommendation:
Please follow the installation recommendations made by the PNO, especially for the simultaneous use of frequency inverters.

Wiring and bus termination

![Diagram of Connector, Wiring, and Bus termination]
2 PROFIBUS-DP description

2.3 PROFIBUS-DP

PROFIBUS-DP is designed for high-speed data exchange at the field level. The central control devices, PLC/PC for instance, communicate through a fast serial connection with distributed field devices such as I/O, paperless recorders and controllers. Data exchange with these distributed devices is mainly cyclic. The communication functions required for this are defined by the basic PROFIBUS-DP functions in accordance with IEC 61158 and IEC 61784.

**Basic functions**

The central controller (master) reads the input information cyclically from the slaves and writes the output information cyclically to the slaves. The bus cycle time must be shorter than the program cycle time of the central PLC. In addition to cyclic user data transmission, PROFIBUS-DP provides powerful functions for diagnostics and commissioning.

<table>
<thead>
<tr>
<th>Transmission technology</th>
</tr>
</thead>
<tbody>
<tr>
<td>• RS485 twisted pair</td>
</tr>
<tr>
<td>• Baud rates of 9.6 kbit/sec up to 12 Mbit/sec</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bus access</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Master and slave devices, max. 126 stations on one bus</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Communication</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Peer-to-peer (user data communication)</td>
</tr>
<tr>
<td>• Cyclic master-slave user data communication</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Operating states</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Operate: Cyclic transmission of input and output data</td>
</tr>
<tr>
<td>• Clear: Inputs are read, outputs remain in secure state</td>
</tr>
<tr>
<td>• Stop: Only master-master data transfer is possible</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Synchronization</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Sync mode: not supported by JUMO devices</td>
</tr>
<tr>
<td>• Freeze mode: not supported by JUMO devices</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Functionality</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Cyclic user data transfer between DP master and DP slave(s)</td>
</tr>
<tr>
<td>• Dynamic activation or deactivation of individual DP slaves</td>
</tr>
<tr>
<td>• Checking the configuration of the DP slaves</td>
</tr>
<tr>
<td>• Address assignment for the DP slaves via the bus</td>
</tr>
<tr>
<td>• Configuration of the DP master via the bus</td>
</tr>
<tr>
<td>• Maximum of 246 bytes input/output data for each DP slave</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Protective functions</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Address monitoring for the DP slaves</td>
</tr>
<tr>
<td>• Access protection for inputs/outputs of the DP slaves</td>
</tr>
<tr>
<td>• Monitoring of user data communication with adjustable monitoring timer in the DP master</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Device types</th>
</tr>
</thead>
<tbody>
<tr>
<td>• DP master Class 2, e. g. programming/project design devices</td>
</tr>
<tr>
<td>• DP master Class 1, e. g. central automation devices such as PLC, PC…</td>
</tr>
<tr>
<td>• DP slave e. g. devices with binary or analog inputs/outputs, controllers, recorders…</td>
</tr>
</tbody>
</table>
Cyclic data transmission

The data transmission between the DP master and the DP slaves is carried out by the master in a defined, recurring order. When configuring the bus system, the user defines the assignment of a DP slave to the master. It is also defined which DP slaves are to be included in, or excluded from, the cyclic user data communication.

Data transmission between the master and the DP slaves is divided into three phases: parameterization, configuration and data transfer. Before a DP slave enters the data transfer phase, the DP master checks in the parameterization and configuration phase whether the planned configuration matches the actual device configuration. In the course of this check, the device type, format and length information, as well as the number of inputs and outputs must agree. These checks provide the user with reliable protection against parameterization errors. In addition to the user data transfer, which is performed automatically by the DP master, new parameterization data can be sent to the DP slaves at the request of the user.
3 Configuring a PROFIBUS-DP system

3.1 The GSD file

Device data (GSD) enable open project design.

PROFIBUS-DP devices have different performance features. They differ with respect to the available functionality (e.g., number of I/O signals, diagnostic messages) or possible bus parameters, such as baud rate and time monitoring. These parameters vary individually for each device type and manufacturer. In order to obtain simple Plug & Play configuration for PROFIBUS-DP, the characteristic device features are defined in an electronic data sheet (device data base file, GSD file). The standardized GSD files expand open communication up to the operator level. By means of the project design tool, which is based on the GSD files, devices from different manufacturers can be integrated into a bus system, simply and user-friendly. The GSD files provide a clear and comprehensive description of a device type in a precisely defined format. GSD files are prepared according to the application. The defined file format permits the project design system to simply read in the device data of any PROFIBUS-DP device and automatically use this information when configuring the bus system. Already during the project design phase, the project design system can automatically perform checks for input errors and the consistency of data entered in relation to the entire system.

The GSD files are divided into three sections.

- **General specifications**
  This section contains information on manufacturer and device names, hardware and software release states and the baud rates supported.

- **DP master-related specifications**
  This section contains all the parameters related to DP master devices only, such as the maximum number of DP slaves that can be connected, or upload and download options. This section is not available for slave devices.

- **DP slave-related specifications**
  This section contains all slave-related specifications, such as the number and type of I/O channels, specifications of diagnostic texts and information on the consistency of I/O data.

The GSD file contains lists, such as the baud rates supported by the device, as well as the possibility of describing the modules available in a modular device.
3 Configuring a PROFIBUS-DP system

3.2 Configuration procedure

**Plug & Play**

To simplify the configuration of the PROFIBUS-DP system, the DP master (PLC) is configured using the PROFIBUS-DP configurator and the GSD files, or in the PLC through the hardware configurator.

**Configuration steps:**

- Create GSD file by using the GSD generator
- Load GSD files of the PROFIBUS-DP slaves into the PROFIBUS-DP network configuration software
- Perform configuration
- Load configuration into the system (e.g. PLC)

---

The characteristic device features of a DP slave are specified by the manufacturer, clearly and comprehensively in a precisely defined format, in the GSD file (Device Data Base File).

This software can read in the GSD files from PROFIBUS-DP devices of any manufacturer and integrate them for the configuration of the bus system.

Already in the project design phase, the PROFIBUS-DP configurator automatically checks the files that have been entered for errors in system consistency.

The result of the configuration is read into the DP master (PLC).
3 Configuring a PROFIBUS-DP system

3.3 The GSD generator

3.3.1 General

GSD files for JUMO devices with a PROFIBUS-DP interface are generated by the user with the aid of the GSD generator. The devices with a PROFIBUS-DP interface that are available from JUMO can send or receive a large variety of variables (parameters). Since, however, in most applications, only a portion of these variables will be sent via PROFIBUS-DP, the GSD generator is used to make a selection of these variables.

After selection of the device, all available variables are shown in the “Parameters” window. Only after these have been copied to the “Input” or “Output” window will they later be contained in the GSD file for processing or preprocessing by the DP master (PLC).

3.3.2 Operation

File menu

Input window
(input for master/PLC)

Output window
(output for master/PLC)

Window with available parameters

Device name for hardware catalog

If different GSD files are required for devices of the same type, this standard name should be altered in such a way as to enable an unambiguous assignment of the Profibus master in the hardware configuration.

These addresses are described in the MODBUS interface description.

⇒ B 70.3041.2

These addresses are described in the MODBUS interface description.
3 Configuring a PROFIBUS-DP system

When using SIEMENS Simatic S7 for project design, the file names in the GSD file must not be longer than 8 characters. GSD files with long file names cannot be entered into the hardware catalog of the PLC.

### File menu

The file menu can be called up by using the Alt-D combination or the left mouse button. It provides the following options:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>New</strong></td>
<td>After calling up the function which creates a new GSD file, the available devices are selected. After selection of the required device, all available parameters are shown in the parameter window.</td>
</tr>
<tr>
<td><strong>Open</strong></td>
<td>This function opens an existing GSD file.</td>
</tr>
<tr>
<td><strong>Save/Save as</strong></td>
<td>This function is available for saving the generated or altered GSD file.</td>
</tr>
<tr>
<td><strong>Diagnosis</strong></td>
<td>Using this function, the GSD file can be tested in conjunction with a PROFIBUS-DP master simulator from B+W and the Profibus slave.</td>
</tr>
<tr>
<td><strong>Print preview</strong></td>
<td>shows the preview of a report(^1), that can be printed.</td>
</tr>
<tr>
<td><strong>Print</strong></td>
<td>prints a report(^1).</td>
</tr>
<tr>
<td><strong>Default settings</strong></td>
<td>The language to be used at the next restart of the program can be selected here.</td>
</tr>
<tr>
<td><strong>Exit</strong></td>
<td>exits the program.</td>
</tr>
</tbody>
</table>

\(^1\) The report contains additional information for the PLC programmer (e.g. data type of the selected parameters).

⇒ Chapter 3.3.3 “Example report”
3.3.3 Example report

I/O Report
Gerät: JUMO dTRON 304/308/316
Länge der Eingänge (Byte): 27
Länge der Ausgänge (Byte): 6

Eingänge

<table>
<thead>
<tr>
<th>Byte</th>
<th>Beschreibung</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Interface-Status</td>
<td>BYTE</td>
</tr>
<tr>
<td>1</td>
<td>Regler\Istwert</td>
<td>REAL</td>
</tr>
<tr>
<td>5</td>
<td>Regler\Sollwert</td>
<td>REAL</td>
</tr>
<tr>
<td>9</td>
<td>Regler\Regelabweichung</td>
<td>REAL</td>
</tr>
<tr>
<td>13</td>
<td>Regler\Stellgradanzeige</td>
<td>REAL</td>
</tr>
<tr>
<td>17</td>
<td>Regler\1.Reglerausgang</td>
<td>REAL</td>
</tr>
<tr>
<td>21</td>
<td>Regler\Schaltstellung 2</td>
<td>INTEGER</td>
</tr>
<tr>
<td>23</td>
<td>Timer\Timererstzeit</td>
<td>LONG</td>
</tr>
</tbody>
</table>

Ausgänge

<table>
<thead>
<tr>
<th>Byte</th>
<th>Beschreibung</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Sollwerte\Sollwert 1</td>
<td>REAL</td>
</tr>
<tr>
<td>4</td>
<td>Regler\Handstelzgrad</td>
<td>INTEGER</td>
</tr>
</tbody>
</table>
3 Configuring a PROFIBUS-DP system

3.3.4 Arrangement of a GSD file

```
GSD file Gateway PROFIBUS-DP
JUMO dTRON 300
Release 08.07.2005

#Profibus_DP
GSD_Revision = 2 ; extended GSD file is supported
Vendor_Name = "JUMO" ; name of the manufacturer
Model_Name = "dTRON 3xx" ; name of the DP device
Revision = "revision level 2.0" ; actual edition of the DP device
Ident_Number = 0x09AB ; exact type designation of the DP device
Protocol_Ident = 0 ; protocol characteristics PROFIBUS-DP
Station_Type = 0 ; DP slave
FMS_supp = 0 ; DP device only
Hardware_Release = "1.00" ; actual edition of the hardware
Hardware_Release = "1.00" ; actual edition of the hardware

; the following baud rates are supported
9.6_supp = 1 ; 9.6 kbaud
19.2_supp = 1 ; 19.2 kbaud
45.45_supp = 1 ; 45.45 kbaud
93.75_supp = 1 ; 93.75 kbaud
187.5_supp = 1 ; 187.5 kbaud
500_supp = 1 ; 500 kbaud
1.5M_supp = 1 ; 1.5 Mbaud
3M_supp = 1 ; 3 Mbaud
6M_supp = 1 ; 6 Mbaud
12M_supp = 1 ; 12 Mbaud

MaxTsdr_9.6 = 60
MaxTsdr_19.2 = 60
MaxTsdr_45.45 = 60
MaxTsdr_93.75 = 60
MaxTsdr_187.5 = 60
MaxTsdr_500 = 100
MaxTsdr_1.5M = 150
MaxTsdr_3M = 250
MaxTsdr_6M = 350
MaxTsdr_12M = 800

Redundancy = 0 ; no redundant transmission
Repeater_Ctrl_Sig = 1 ; Plug signal CNTR-P TTI level
24V_Pins = 0 ; Plug signals M24V and P24 V not connected
Implementation_Type = "SPC3" ; Application of ASIC SPC3
```

3 Configuring a PROFIBUS-DP system

;; ;*** Slave specific values ***
Freeze_Mode_supp = 0 ; Freeze mode is not supported
Sync_Mode_supp = 0 ; Sync mode is not supported
Auto_Baud_supp = 1 ; Automatic recognition of baud rate
Set_Slave_Add_supp = 1 ; Set_Slave_Add is supported
Min_Slave_Interval = 6 ; Slave interval = 0.6 ms
Modular_Station = 1 ; Modular station
Max_Module = 4
Max_Diag_Data_Len = 6
Slave_Family = 5

;; ;*** Parameterization ***
; These lines are for locating PBC file, and initial data length.
; Do not disturb!!!
atPBC_File = C:\PROGRAMS\JUMO\GSDGEN\14401XX\D\ju_d300.PBC
atINIT_LEN = 2

User_Prm_Data_Len = 16
User_Prm_Data = 0x00, 0x03, 0x02, 0x01, 0x13, 0x00, 0x43, 0x04, 0x17, 0x20, 0x10, 0x08, 0x27, 0x20, 0x00, 0x08
Max_Input_Len = 13
Max_Output_Len = 8
Max_Data_Len = 21

;===============================================================================
Module = "Interface Mode" 0x10
Preset = 1
Endmodule

Module = "Controller/Process Value" 0x13
Preset = 1
Endmodule

Module = "acycl. Data/Block-Read" 0x17
Preset = 1
Endmodule

Module = "acycl. Data/Block-Write" 0x27
Preset = 1
Endmodule
;===============================================================================

The GSD file has been arranged for installation on the SIMATIC S7 (SIEMENS).

Should installation problems occur with other controls, all entries Preset=1 must be deleted.

In this case, it is additionally necessary to set up the variables selected in the GSD generator in the correct sequence in the process image of the PLC.
3 Configuring a PROFIBUS-DP system

Select parameter

If an existing file has been opened, or a new one created, all available parameters are shown in the parameter window.

These addresses are described in the MODBUS interface description. ⇒ B 70.3041.2

Device name for hardware catalog

If different GSD files are required for devices of the same type, this standard name should be altered in such a way as to enable an unambiguous assignment of the Profibus master in the hardware configuration.

A click with the left mouse button on the “+” (Regler) or “−” (Solwerte) symbol will extend the parameter list or reduce it.

Click on the parameter with the left mouse button, and, keeping it pressed, copy it to the input or output window by Drag & Drop.

Remove parameter

Parameters are deleted from the input or output window by using the left-pointing arrow ←.

The parameter “Interface status” will automatically appear in the input window and cannot be deleted. It is used for diagnosis of the internal data transmission in the device and should be evaluated by the PLC program, to make sure the data are valid.

0 : internal communication in device is OK
unequal 0 : faulty communication in device
3 Configuring a PROFIBUS-DP system

3.4 Connection example

The example below shows the path for the connection of a JUMO controller (dTRON 3XX) to a SIMATIC S7.

3.4.1 dTRON 3XX

* Connect the device to the PLC.
* Set the device address.
  The device address can be selected via the instrument keys or through the setup program.

3.4.2 JUMO GSD generator

* Start up the GSD generator (Example: Start ➔ Programs ➔ JUMO devices ➔ PROFIBUS ➔ JUMO GSD generator).
* First select the device.

* In the left window, select the variable that is to be transmitted to the DP master
* Click on the direction arrow and the variable appears in the right window, or move to the right window by Drag&Drop.
3 Configuring a PROFIBUS-DP system

* Save the GSD file in any folder.

When using SIEMENS Simatic S7 for project design, the file names in the GSD file must not be longer than 8 characters.

3.4.3 PLC configuration

* Start the PLC software.

* Call up the hardware configuration and execute the menu command “Install new GSD”.

The new GSD file is read in and processed, and the controller is inserted into the hardware catalog.
3 Configuring a PROFIBUS-DP system

* Open the hardware catalog and place the new device in the working area.

The controller is placed on the bus using the left mouse button. After releasing the mouse button, the controller address has to be assigned.

Via the GSD file of the slaves, the master receives information on which baud rates are supported.
Finally, you have to load the configuration into the PLC (Target system ➔ Download to module)

If a device with PROFIBUS-DP interface is operated on a master system (PLC), suitable error analysis routines should be provided on the master side.

In conjunction with a SIMATIC S7, the OB86 should be installed in the PLC, so that failure of a PROFIBUS-DP device can be detected, analyzed and registered for the specific plant.

The parameter “Interface status” will automatically appear in the input window and cannot be deleted. It is used for diagnosis of the internal data transmission in the device and should by evaluated by the PLC program, so that, for instance, a communication problem within the device can be detected by the PLC master.

0 : internal communication in device is OK
unequal 0 : faulty communication in device
When using JUMO devices in a PROFIBUS-DP system, please take note of their data format.

**Integer values**

Integer values are transmitted in the following format:
- first the high byte,
- then the low byte.

**Float values/real values**

The float/real values of the cyclic data for the device are transmitted using the IEEE-754 standard format (32 bits).

The float/real values for the controllers/generators and for the acyclic data of the device are transmitted in the MODbus format.

The IEEE-754 standard format and the MODbus format differ in the transmission sequence of the individual bytes. In the MODbus format bytes 1 and 2 are swapped with bytes 3 and 4 (first the high byte, then the low byte).

**Single-float format (32 bits) as per IEEE 754 standard**

<table>
<thead>
<tr>
<th>SEE</th>
<th>EMMM</th>
<th>MMMM</th>
<th>MMMM</th>
</tr>
</thead>
<tbody>
<tr>
<td>byte 1</td>
<td>byte 2</td>
<td>byte 3</td>
<td>Byte 4</td>
</tr>
</tbody>
</table>

- S - sign bit (bit31)
- E - exponent in complement to base 2 (bit23 — bit30)
- M - 23-bit normalized mantissa (bit0 — bit22)

Example:
calculation of the real number from sign, exponent and mantissa.

byte1 = 40h, byte2 = F0, byte 3 = 0, byte 4 = 0
40F00000h = 0100 0000 1111 0000 0000 0000 0000 0000b

S = 0
E = 100 0000 1
M = 111 0000 0000 0000 0000 0000 0000 0000

Value = \(-1^S \cdot 2^{\text{exponent}-127} \cdot (1 + M_{b22} \cdot 2^{-1} + M_{b21} \cdot 2^{-2} + M_{b20} \cdot 2^{-3} + M_{b19} \cdot 2^{-4} + \ldots)\)

Value = \(-1^0 \cdot 2^{129-127} \cdot (1 + 1 \cdot 2^{-1} + 1 \cdot 2^{-2} + 1 \cdot 2^{-3} + 0 \cdot 2^{-4})\)

Value = 1 \cdot 2^2 \cdot (1 + 0.5 + 0.25 + 0.125 + 0)

Value = 1 \cdot 4 \cdot 1.875

Value = 7.5
## 4 Data format of the JUMO devices

### MODbus float format

<table>
<thead>
<tr>
<th>Address x</th>
<th>Address x+1</th>
</tr>
</thead>
<tbody>
<tr>
<td>MMMMMMMMM</td>
<td>MMMMMMMMM</td>
</tr>
<tr>
<td>byte 3</td>
<td>byte 4</td>
</tr>
<tr>
<td></td>
<td>SEEEEEE</td>
</tr>
<tr>
<td></td>
<td>EMMMMMMM</td>
</tr>
<tr>
<td>byte 1</td>
<td>byte 2</td>
</tr>
</tbody>
</table>

After/before the transmission from/to the device, the bytes of the float value have to be swapped accordingly.

Many compilers (e.g. Microsoft C++, Turbo C++, Turbo Pascal, Keil C51) store the float values in the following order (Intel compatibility):

### float value

<table>
<thead>
<tr>
<th>Storage address x</th>
<th>Storage address x+1</th>
<th>Storage address x+2</th>
<th>Storage address x+3</th>
</tr>
</thead>
<tbody>
<tr>
<td>MMMMMMMMM</td>
<td>MMMMMMMMM</td>
<td>EMMMMMMM</td>
<td>SEEEEEE</td>
</tr>
<tr>
<td>byte 4</td>
<td>byte 3</td>
<td>byte 2</td>
<td>byte 1</td>
</tr>
</tbody>
</table>

Please check how float values are stored in your application. If necessary, the bytes have to be swapped accordingly.
5.1 Connection for dTRON 3XX

* Determine the position of the PROFIBUS-DP interface from the type code (see Operating Instructions B70.3041.0)

**Rear view**

In this example, the PROFIBUS-DP interface is in option slot 1

**Assignment of the 9-pole D-SUB socket**

<table>
<thead>
<tr>
<th>COM2 D-Sub socket</th>
<th>Pin: signal on device terminal strip 1</th>
<th>Designation</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>1: VP</td>
<td>Supply voltage positive</td>
</tr>
<tr>
<td>3</td>
<td>2: Rx/D/Tx-D-P</td>
<td>Receive/Transmit Data positive</td>
</tr>
<tr>
<td>8</td>
<td>3: Rx/D/Tx-D-N</td>
<td>Receive/Transmit Data negative</td>
</tr>
<tr>
<td>5</td>
<td>4: DGNd</td>
<td>Ground</td>
</tr>
</tbody>
</table>
5 Device-specific data

5.2 Setting the slave address

The slave address is set at the configuration level:

*Configuration level ➔ Interfaces ➔*

**PROFIBUS-DP**

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Value/selection</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P_{ack}$</td>
<td>0</td>
<td>Intel</td>
</tr>
<tr>
<td>1</td>
<td>Motorola</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Intel integer</td>
<td></td>
</tr>
</tbody>
</table>

**Device address**

| $R_{dr}$ | 0 — 128 — 255 | Address in data network |

**Analog marker**

| $R_{an}R_{p}$ | -1999 to 0 to +9999 | Analog value |

**Binary marker**

| $B_{b}n_{p}$ | 0 — 255 | Binary value |

Factory settings are shown **bold**.

The baud rate is determined automatically (max. 12Mbps).

5.3 Diagnostic and status messages

If errors occur during communication with the device, the error message “ProF” appears in the display.

Please check the wiring and the master (PLC).

It may be necessary to restart the system.

The error message can be suppressed by setting the slave address 0.

5.4 Acyclic data transmission

You can use acyclic data transmission to read and write a large number of the parameter/measurement and process data for the controller that are documented in the Interface Description B 70.3041.2

The acyclic data are also transmitted with the cyclic data transfer (DPV0).

In order to establish communication with the device, 4 info bytes and a maximum of 4 bytes of user data must be transmitted. Communication by means of acyclic data is mainly through the well-known MODbus communication, i.e. the mechanisms of MODbus communication, such as the function (CMD) and address are used in this case.

The advantage of the acyclic transmission method is that it is possible to break through the limits imposed by the PLC, such as a maximum of 128 bytes of input and output data in the process image of the SIMATIC S7-300, or...
5 Device-specific data

a maximum of 31 module entries (number of cyclic parameters) in the GSD file. Any number of parameters can be transmitted one after another and then processed.

The disadvantage of acyclic transmission is that an additional interface must be integrated into the PLC, to ensure the operation of the transmission procedure described below.

On the installation CD supplied with the PROFIBUS interface, you will find a PLC demo program (Dtron3xx.zip) for the Siemens SIMATIC S7, using the widely-used CPU 315-2 DP. JUMO can only make this demo program available for the SIMATIC S7. JUMO does not provide any guarantee that the program, which is intended to facilitate the first commissioning of the acyclic transmission method, will function without errors in all applications.

Selecting the acyclic transmission method in the GSD generator:

Both entries for acyclic data transmission (Block Read and Block Write) lie within the consistent area of the process image of the PLC.
5 Device-specific data

5.4.1 Arrangement of the protocol

<table>
<thead>
<tr>
<th>Byte no.</th>
<th>1 – 2</th>
<th>3 – 4</th>
<th>5 – 8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field</td>
<td>Control word</td>
<td>MODbus address</td>
<td>Data</td>
</tr>
<tr>
<td>Contents</td>
<td>Control</td>
<td>Length</td>
<td>Function</td>
</tr>
<tr>
<td>4-bit (bits 7 – 4)</td>
<td>4-bit (bits 3 – 0)</td>
<td>1 byte</td>
<td></td>
</tr>
<tr>
<td>Job OK</td>
<td>Job error</td>
<td>Job Toggle 2</td>
<td>Job Toggle 1</td>
</tr>
<tr>
<td>0x11 read INT</td>
<td>0x13 read FLOAT</td>
<td>0x21 write INT</td>
<td>0x23 write FLOAT</td>
</tr>
</tbody>
</table>

**Control bits 4 – 5**

Job Toggle 1, Job Toggle 2

These two bits are required to control the interaction between the PLC and the device. Control bits 4 and 5 may only be set after the transmit buffer has been completely filled up. In order to ensure that the correct data are evaluated and processed, the following procedure must be adhered to.

<table>
<thead>
<tr>
<th>Bit 5</th>
<th>Bit 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0</td>
<td>Job 1 is mirrored by the device.</td>
</tr>
<tr>
<td>0 1</td>
<td>Bit 4 is set, job 1 is processed for the first time.</td>
</tr>
<tr>
<td>1 0</td>
<td>Bit 5 is set, job 1 is processed again.</td>
</tr>
<tr>
<td>0 0</td>
<td>Job 2 is mirrored by the device.</td>
</tr>
<tr>
<td>0 1</td>
<td>Bit 4 is set, job 2 is processed for the first time.</td>
</tr>
<tr>
<td>...</td>
<td>..............................................................</td>
</tr>
</tbody>
</table>

The internal design of the device means that it is mandatory to adhere to the procedure above. Control bits 6 – 7

**Control bits 6 – 7**

Job OK, job error

Bit 6 and Bit 7 are signals to the PLC, when the telegram has been evaluated by the device, and the PLC can generate and transmit the next command for the device.
5 Device-specific data

Sequence: Case 1, everything functioning OK

<table>
<thead>
<tr>
<th>Bit 7</th>
<th>Bit 6</th>
<th>Bit 5</th>
<th>Bit 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Sequence: Case 2, not everything functioning OK

<table>
<thead>
<tr>
<th>Bit 7</th>
<th>Bit 6</th>
<th>Bit 5</th>
<th>Bit 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Sequence: Case 3, not everything functioning OK

<table>
<thead>
<tr>
<th>Bit 7</th>
<th>Bit 6</th>
<th>Bit 5</th>
<th>Bit 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

Address

A large number of the addresses listed in the Interface Description B 70.3590.2 can be transmitted through the acyclic transmission procedure. The transmission follows the well-known MODbus communication very closely.

User data

A maximum of 4-bytes of user data can be transmitted. The quantity of user data used is stored in the function byte.
5 Device-specific data

An example will clarify the basic sequence to be used for the data transmission between the PLC and the device. In this case, the controller setpoint = 50.0 is defined by the PLC for Controller 1.

<table>
<thead>
<tr>
<th>Byte no.</th>
<th>1 — 2</th>
<th>3 — 4</th>
<th>5 — 8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field</td>
<td>Control word</td>
<td>Modbus address</td>
<td>Data</td>
</tr>
<tr>
<td>Contents</td>
<td>Control</td>
<td>Length</td>
<td>Function</td>
</tr>
</tbody>
</table>

3.) PLC transmits telegram with the information Toggle 1.

4.) Device evaluates the telegram, and sends back Job OK or Job error.

5.) If the PLC receives Job OK, the telegram is sent again to the device, with Toggle 2. If Job error is received, processing can be canceled immediately, since there is an error in the telegram layout.

6.) Device evaluates the telegram again, and sends back Job OK or Job error.
5.4.2 Demo program dTRON3xx.zip

The demo program for the controllers can be found on the installation CD. JUMO can only make it available for the SIMATIC S7. The CPU 315-2 DP is used.

You can find a program listing in PDF format on the installation CD, by first calling up the installation program, and then clicking on “Documentation ➔ dTRON3xx_azyk.pdf”. JUMO does not provide any guarantee that the program, which is intended to facilitate the first commissioning of the acyclic transmission method, will function without errors in all applications.

The procedure for data communication is controlled by markers. The demo program processes a total of 6 commands, in the course of which Controller/process value and Controller setpoint are read, and Setpoints/setpoint1 is written and transmitted to 2 dTRON3XX.
5 Device-specific data

OB1
The operating system of the PLC processes OB1 cyclically. If the processing of OB1 is finished, then the operating system starts processing OB1 again. The cyclic processing of OB1 is started when the start-up has finished.

OB86
The operating system of the PLC calls up OB86 if it detects the failure of an expansion device, a DP master system or a station in the decentralized periphery (both for incoming and outgoing events). If such an error occurs, and OB86 is not present, then the CPU goes to the STOP status. The demo program only evaluates the incoming events for error codes 0xC4 and 0xC5, by incrementing marker word 28.

FC1:
Function for reading 4 analog values and writing 2 analog values into the SIMATIC S7 - 300 (315 - 2 DP). The reading and writing of process values is carried out separately, and can be started separately. Reading is started once through the marker 30.3 StartTransferZyklisch, i.e. all 4 analog values are read, one after another, and then the procedure is terminated. The marker 30.1 StartTransferAzyklisch can be used to separately start the writing of the 2 analog values. The markers 30.0 SteuerFlagZyklisch and 30.2 SteuerFlagAzyklisch indicate the duration of the transmission separately for reading and writing.

The command processing is set up in a sequential list, i.e. if one command has been processed, the next is started automatically.

FC10:
Operates the PROFIBUS-DP interface (driver). The function has some defined transfer parameters, which are explained in the following text.

Addr:
Address for the acyclic data in the process image of the PLC. The addresses for the acyclic data in the input process image and the output process image must be identical. It is not possible to use different addresses. The address must be transferred to FC 10 in HEX format.
5 Device-specific data

Command: Command syntax that is to be sent to the device, e.g. Controller 1 set Setpoint = 50.0, .... The program is laid out so that a pool of possible commands is predefined in a data block (DB).

Response: The response that the device sends to the PLC is also saved in a data block, and can be interpreted and evaluated by other PLC program components.

STRT: Start pulse, i.e. the start of command processing. The telegram that is transferred under Command is sent to the device.

RDY: Command processing finished. The device has sent back a response to the PLC, after which the RDY flag is set by FC10 in order to indicate to the higher-level processing that the execution of the telegram has been concluded. The RDY flag will also be set after a time-out error.

Toggle: The Toggle flag is reset by FC10. The toggle flag that is to be transferred is interpreted and the toggle information is prepared for the device.

DB10: The response from the device is placed in this data block. At present, 6 different response messages can be stored. The size of DB10 can be adjusted at any time. DB10 saves the complete response telegram from the controller, i.e. the user data as well as the control information and the address.

DB20: Several predefined commands are stored in Data Block 20. These can be used individually in this demo program. For each command, there is a short description of the data that can be transferred. You can alter DB20 or add other commands at any time. The necessary information can be found in the Operating Instructions B 70.3590.2.
5 Device-specific data

**Command 000:**
Read Controller\process value

**Command 001**
Read Controller\setpoint

**Command 002**
Read Controller\controller output 1

**Command 003**
Read Controller\controller output 2

**Command 004**
Write Setpoints/setpoint1

**Command 005**
Write Controller\manual output

**UDT10:**
Universal Data Type. The organization of the input and output data channel for the device is defined in UDT10.

**VAT1:**
The variable table that is set up makes it possible to check the data communication or control the command processing.
5.4.3 Block diagram of the demo program

0:OB1 is automatically and cyclically executed by the CPU of the PLC.

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1)</td>
<td>FC1 is also called up cyclically by the PLC. If the marker 30.3 StartTransferZyklisch is set, the result is that processing of the command starts in FC1, whereby the first command is prepared by FC10 and sent to the controller.</td>
</tr>
<tr>
<td>2)</td>
<td>FC10 loads the required command from DB20 and prepares the corresponding toggle information.</td>
</tr>
<tr>
<td>3)</td>
<td>The data telegram is entered into the process image of the PLC by FC10, and thus transmitted to the controller.</td>
</tr>
<tr>
<td>4)</td>
<td>The controller processes the data telegram that has been received, and makes a response message available for FC10 in the process image of the PLC.</td>
</tr>
<tr>
<td>5)</td>
<td>FC10 evaluates the response by comparing it with the transmitted telegram, and saving it in DB10 for the following PLC program.</td>
</tr>
</tbody>
</table>
5 Device-specific data

6) FC1 receives a signal, through the RDY flag, that the processing of a command has been concluded. When this information reaches FC1, the next command is immediately dispatched to the JUMO dTRON 3XX.

7) The data that is collected or is to be sent to the JUMO dTRON 3XX is, in practice, evaluated or handled by other PLC program functions. As a consequence, it is always necessary to check whether the data are correct before they can be interpreted. The two control flags in the telegram header (Job OK and Job error) must be checked.

1— 8 Processing sequence

⚠️ JUMO cannot provide any guarantee that the components or extensions listed above will ensure fault-free operation of the system, since any error handling must always be defined on a system-specific basis.

5.4.4 Timing of acyclic data

Acyclic data transmission enables universal access to a large number of the data and parameters that are accessible via MODbus (without alteration to the configuration or program). However, because of the larger number of processing steps, it requires a longer time to update.

5.4.5 Commands (GSD generator)

If you require additional information on the commands within the GSD generator, please refer to the Interface Description B 70.3041.2. The GSD file has been laid out for installation on the SIMATIC S7. Should installation problems occur with other controls, all entries Preset=1 must be deleted.

☞ Chapter 3.3.4

In this case it will also be necessary to set up the variables selected in the GSD generator in the correct sequence in the process image of the PLC.