



***DP - Driver of PROFIBUS DP Network
Protocol for PROFiboard Board
User's Manual***

Doc. No. ENP4016
Version: 29-08-2005

ASKOM[®] and **asix**[®] are registered trademarks of ASKOM Spółka z o.o., Gliwice. Other brand names, trademarks, and registered trademarks are the property of their respective holders.

All rights reserved including the right of reproduction in whole or in part in any form. No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage and retrieval system, without prior written permission from the ASKOM.

ASKOM sp. z o. o. shall not be liable for any damages arising out of the use of information included in the publication content.

Copyright © 2005, ASKOM Sp. z o. o., Gliwice



ASKOM Sp. z o. o., ul. Józefa Sowińskiego 13, 44-121 Gliwice,
tel. +48 (0) 32 3018100, fax +48 (0) 32 3018101,
<http://www.askom.com.pl>, e-mail: office@askom.com.pl

1. DP - Driver of PROFIBUS DP Network Protocol for PROFIboard Board

1.1. Driver Use

The DP driver is used for data exchange in the PROFIBUS network with devices operating according to the PROFIBUS DP standard. The **asix** system computer must be provided with the PROFIboard NT communication processor board and with the PROFIboard NT v 5.20 software package of Softing GmbH.

1.2. Declaration of Transmission Channel

The full syntax of declaration of transmission channel working according to the DP protocol is given below:

logical_name=DP,card_no, address,section_name

where:

board_no - number of the PROFIboard NT board which is used for communication with a given device (DP slave). In the present version the DP driver can control only one PROFIboard NT board;

address - address assigned to the DP device;

section_name - name of section in the application INI file, where the configuration parameters of the given DP device are contained.

The DP driver is loaded as a DLL automatically.

1.3. Configuration of DP Devices

Configuration parameters of each DP device (DP slave) should be declared in a separate section of the application INI file. These parameters should be accessible in the device specification or in the GSD file supplied by the manufacturer with the device. The section name is assigned by the user and must be unique in the application INI file.

The individual configuration parameters are transferred in separate items of the section. Each item is as follows:

parameter=value

where:

parameter - type of configuration parameter;
value - parameter value (in decimal format !)

Types of configuration parameters, which **must be transferred** by the application designer are given below:

IDENT_NUMBER	device identifier	(PNO ident. number)
GROUP_IDENT	group identifier	(group member bits)
USER_PRM_DATA		(prm_user_data)
.. variable number of USER_RPM_DATA items		

USER_PRM_DATA	(prm_user_data)
MODULE_ID	(cfg_data)
.. variable number of MODULE_ID items	
MODULE_ID	(cfg_data)
<i>aat_data</i> parameters are optional and if not declared they are set to 0.	
NUMBER_OF_INPUTS	(aat_data)
NUMBER_OF_OUTPUTS	(aat_data)
OFFSET_OF_INPUTS	(aat_data)
OFFSET_OF_OUTPUTS	(aat_data)
<i>slave_data</i> parameters are optional and if not declared they are omitted.	
SLAVE_DATA	(slave_user_data)
.. variable number of SLAVE_DATA items	
SLAVE_DATA	(slave_user_data)

1.4. Addressing the Process Variables

The values transferred from modules connected to the DP device are written to an I/O buffer of the DP driver, in order resulting from arrangement of Input/Output modules in the cassette of the DP device. The addressing process variables requires an indication of:

- buffer type (input buffer or output buffer);
- byte no. (in the buffer), where the value of a given input/output is stored; depending on the type of process variable, the variable value occupies one byte (one-byte type variable) or 2 successive bytes (2-byte type variable);
- type of the variable (one-byte or 2-byte).

The syntax of symbolic address which is used for variables belonging to the DP driver channel is as follows:

<type><index>

where:

<i>type</i>	- type of process variables:
IB	- byte from the input buffer,
IW	- 2 successive bytes from the input buffer treated as a fixed-point unsigned number in INTEL format,
IDW	- 4 successive bytes from the input buffer treated as a double word in INTEL format,
IFP	- 4 successive bytes from the input buffer treated as a floating-point number in INTEL format,
IWM	- 2 successive bytes from the input buffer treated as a fixed-point unsigned number in MOTOROLA format,
IDWM	- 4 successive bytes from the input buffer treated as a double word in MOTOROLA format,
IFPM	- 4 successive bytes from the input buffer treated as a floating-point number in MOTOROLA format,
OB	- byte from the output buffer,
OW	- 2 successive bytes from the output buffer treated as a fixed-point unsigned number in INTEL format,
ODW	- 4 successive bytes from the output buffer treated as a double word in INTEL format,

OFP	- 4 successive bytes from the output buffer treated as a floating-point number in INTEL format,
OWM	- 2 successive bytes from the output buffer treated as a fixed-point unsigned number in MOTOROLA format,
ODWM	- 4 successive bytes from the output buffer treated as a double word in MOTOROLA format,
OFPM	- 4 successive bytes from the output buffer treated as a floating-point number in MOTOROLA format;
<i>Index</i>	- number of the byte in the input/output buffer.

EXAMPLE

IB9	- 9-th byte from the area of inputs
IW2	- word created from the 2-nd and 3-rd byte of the input area (INTEL format)
IWM2	- word created from the 3-rd and 2-nd byte of the input area (MOTOROLA format)
IDW5	- double word created from the 5-th, 6-th, 7-th and 8-th byte of the input area (INTEL format)
IDWM5	- double word created from the 8-th, 7-th, 6-th and 5-th byte of the input area (MOTOROLA format)

1.5. Driver Configuration

The DP protocol driver may be configured with use of information contained in **[DP]** section placed in the application INI file. Items in the DP section have the following syntax:

item_name = [*number* [,*number*]] [*text*][*YES/NO*]



TRANSMISSION_SPEED=network_no,baud_id

Meaning	- the item is used to declare the transmission speed in the PROFIBUS DP network.
Default value	- by default, the transmission speed is set to 1,5 MB.
Parameter:	
<i>network_no</i>	- number of the PROFIBUS DP network (in the present version always set to 1),
<i>baud_id</i>	- identifier of transmission speed in a PROFIBUS DP network:
	0 - 9,6 kB
	1 - 19,2 kB
	2 - 93,75 kB
	3 - 187,5 kB
	4 - 500 kB
	5 - 750 kB
	6 - 1,5 MB
	7 - 3 MB
	8 - 6 MB
	9 - 12 MB
	11 - 45,45 kB

EXAMPLE

A declaration of transmission speed of 750 kB:

TRANSMISSION_SPEED = 1, 5



REFRESH_CYCLE=number

Meaning	- the item is designed to declare an interval between successive data readings from buffers of the PROFIBoard NT board to the structures of the DP driver.
Default value	- by default, the DP driver reads data from buffers of the PROFIBoard NT board every 0.5 second.
Parameter:	
<i>number</i>	- number of 0.5-second intervals, which must pass between successive data readings from buffers of the PROFIBoard NT board.

EXAMPLE

A declaration of data reading every 1 second:

REFRESH_CYCLE=2



CONSOLE=YES/NO

Meaning	- the item allows creating a console window where the DP driver messages, concerning the status of communication between an asix system computer and DP devices, are displayed.
Default value	- by default, the console window is not created.



LOG_FILE=file_name

Meaning	- the item allows to define a file where all messages of the DP driver, concerning to the status of communication between an asix system computer and DP devices, will be written. If the item does not define the full path, then the log file is created in the current directory.
Default value	- by default, the log file is not created.

EXAMPLE

An example item declaring a transmission channel using the DP protocol for the communication with ET200U no. 7 unit is given below. The following input/output boards are connected to the ET200U (in order of their arrangement in the list):

- Digital Output module (8 outputs) 6ES5 461-8MA11 (identifier 32),
- Analog Input module (4 inputs) 6ES5 464-8ME11 (identifier 83),
- Digital Input module (8 inputs) 6ES5 431-8MA11(identifier 16).
- Digital Output module (8 outputs) 6ES5 461-8MA11 (identifier 32).
- *CHAN1=DP,1,7,SIEM8008*

The transmission channel named CHAN1 has the following parameters defined:

- DP protocol,
- communication by means of the PROFIBUS NT board with the number of 1,
- DP device has number 7 in the PROFIBUS DP network,
- configuration parameters of DP device are contained in the section SIEM8008.

The content of the section [SIEM8008] defining the sample configuration of the ET200U is as follows (all the values are decimal):

```
[SIEM8008]
IDENT_NR=32776
GROUP_IDENT=0
USER_PRM_DATA=0
MODULE_ID=32
MODULE_ID=83
MODULE_ID=32
MODULE_ID=16
```

In the configuration under consideration the area of inputs has 9 bytes. The meaning of the bytes is as follows:

bytes 1,2	- analog input 1
bytes 3,4	- analog input 2
bytes 5,6	- analog input 3
bytes 7,8	- analog input 4
byte 9	- digital input byte

In the configuration under consideration the area of outputs has 2 bytes. The meaning of the bytes is as follows:

byte 1	- digital output byte (the first module 6ES5 451-8MA11)
byte 2	- digital output byte (the second module 6ES5 451-8MA11)

EXAMPLE

Example declarations of process variables are given below:

```
# X1 – digital output – 1-st byte of output buffer
X1, OB1, CHAN1, 1, 1, NOTHING_BYTE
# X2 – digital output - 2-nd byte of output buffer
X2, OB2, CHAN1, 1, 1, NOTHING_BYTE
# X3 – digital input - 9-th byte of input buffer
X3, IB9, CHAN1, 1, 1, NOTHING_BYTE
# X4 – analog input 1 - 1-st and 2-nd bytes of input buffer
X4, IW1, CHAN1, 1, 1, NOTHING
# X5 – analog input 2 - 3-rd and 4-th byte of input buffer
X5, IW3, CHAN1, 1, 1, NOTHING
# X6 – analog input 3 - 5-th and 6-th byte of input buffer
X6, IW5, CHAN1, 1, 1, NOTHING
# X7 – analog input 4 - 7-th and 8-th byte of input buffer
X7, IW7, CHAN1, 1, 1, NOTHING
```


1.	DP - DRIVER OF PROFIBUS DP NETWORK PROTOCOL FOR PROFIBOARD BOARD	3
1.1.	DRIVER USE	3
1.2.	DECLARATION OF TRANSMISSION CHANNEL	3
1.3.	CONFIGURATION OF DP DEVICES	3
1.4.	ADDRESSING THE PROCESS VARIABLES	4
1.5.	DRIVER CONFIGURATION	5