Protocol manual

PROFIBUS-DP master

Hilscher Gesellschaft für Systemautomation mbH
Rheinstraße 78
D-65795 Hattersheim
Germany

Tel. +49 (0) 6190 / 9907 - 0
Fax. +49 (0) 6190 / 9907 - 50

Vertieb: +49 (0) 6190 / 9907 - 0
Hotline und Support: +49 (0) 6190 / 9907 - 99

e-mail: hilscher@hilscher.com
Homepage: http://www.hilscher.com

Index	Date	Version	Chapter	Revision
1	13.05.96	1.000	all	Translation of the german manual (also index 1)
2	13.06.96	1.000	all	Spelling correction of english translation
3	21.11.96	from 1.012	all	Revision of the state tables
4	03.02.97	from 1.016	Diagnostics	change of online state descriptions
5	18.08.97	from 1.105	Diagnostics	change and extension of online state descriptions
6	25.10.99		Technical Data	New chapter Technical Data

Although this protocol implementation has been developed with great care and intensively tested, Hilscher Gesellschaft für Systemautomation mbH cannot guarantee the suitability of this protocol implementation for any purpose not confirmed by us in writing.

Guarantee claims shall be limited to the right to require rectification. Liability for any damages which may have arisen from the use of this protocol implementation or its documentation shall be limited to cases of intent.

We reserve the right to modify our products and their specifiactions at any time in as far as this contributes to technical progress. The version of the manual supplied with the protocol implementation applies.

1 Introduction to the system

The processing of closed loop and open loop control information for machinery and systems with the aid of powerful microchips is widespread in modern automation technology. As a rule, such intelligent units of an automation complex are connected together by one or more communications networks. Within the production facility as a whole, the networks are incorporated in various hierarchical levels, starting with the administration network for the works and ranging up to the network of sensors and actuators in the manufacturing process.

The growing degree of automation and the demand for more and more decentralised units has caused a considerable increase in the quantity of data exchanged in all areas. The establishment of a new concept to accommodate this large flow of data therefore appeared unavoidable.

The lowest hierarchical level in particular, that of sensors and actuators, required the industry to establish a communications network in which the data generated there simultaneously could be transmitted qualitatively and quantitatively as rapidly as possible to the higher level process without the already time critical process sequences noticeably suffering from the transmission speed of the bus.

In the search for a suitable bus system in the field bus area, the parallel transmission process in which each sensor or actuator is connected individually to the higher level controller in a network with star topography was found unsuitable as soon as cost effectiveness calculations revealed that the costs of installation and maintenance exceeded the benefits derived from rapid data transfer.

The only solution here was single cable transmission with serial bus systems, meeting the modern demands for flexible process adaptation and reconfiguration or system expansions.

2 PROFIBUS-DP fundamentals

2.1 The requirements

The connection between the decentralized process sequence and the centralized control system established by the communications system takes place on the lowest hierchical level via the field or process bus.
On this level, the primary requirement is that for a simple protocol sequence and short data transfer times in communications. This guarantees the shortest possible system reaction time to dynamic peripheral conditions.

Together with classical I/O data exchange, acyclical transmission of parameter, diagnostic and configuration data must be possible without decisively impeding the real time capabilities of the bus. Only in this way can a good diagnostic concept be fulfilled and operational reliability be ensured.

2.2 Characteristics

The main function of the PROFIBUS-DP is the cyclical transmission of process data from the control system to the peripherals and in the reverse direction. Access is by the master-slave principle. A master controls the assigned slave devices on the bus in polling operation. Data exchange is initiated by a polling message and terminated by an acknowledgment message from the slave addressed. Each slave therefore only becomes active when required by the master. Simultaneous bus access is thus avoided.

The hybrid access method of PROFIBUS permits combined operation of several bus masters and even mixed operation of PROFIBUS-DP and PROFIBUS-FMS within a single bus section. This however depends on the correct configuration of the bus system and unequivocal assignment of the slave devices to the masters.
The PROFIBUS-DP distinguishes between two types of masters. The class 1 master performs cyclical transmission of user data and provides the application data. The class 1 master can be addressed by a class 2 master for certain functions. Direct access to slaves is not permitted. The functions are limited to support services such as reading of diagnostic informatoin from slaves. A class 2 master is therefore also regarded as a programming or diagnostic device.

2.3 Scope of master-slave functions

The scope of functions of a PROFIBUS-DP master of class 1 comprises the automatic cyclical transmission of the input and output data. This requires successful parameterization and configuration of the slaves.

Within the parameterization phase, each slave is parameterized with the slave specific bus parameters and the monitoring times. The configuration phase then follows. In this phase, the master has the slaves compare the stored specified slave configuration data with the actual configuration data. Only when both phases have been successfully completed is a slave included in the transfer of user data. One exception in the master-slave functions is a global function command with which all slaves assigned to a master can be addressed within one bus access. During parameterisation, the slaves can be assigned to one of eight possible groups, and therefore the command can also be used relative to groups. This multicast command can be used in PROFIBUS-DP for example to control the synchronization of the outputs and inputs or to reset the module outputs to a secure condition.

2.4 Protection functions

The PROFIBUS-DP is equipped with numerous protection functions. These ensure secure communications particularly in the harsh environment of the decentralized peripherals, not only in fault-free operation, but also with external interference or the failure of individual stations.

- Incorrect parameterization is avoided by excluding stations with incorrect parameters from user data operations.
- The failure of stations is registered by the master and displayed to the user by group diagnostic.
- The failure of the transmission line is detected by the slaves with a timed watchdog function, and leads to shutdown of the outputs.
- EMC interference is almost completely filtered out by the particularly interference free transmission process to RS485 with the differential signal.
- Data transfer errors are detected by the CRC test and lead to repetition of the message.

2.5 Commissioning

Before a PROFIBUS-DP system can be put in operation, all the connected stations including the master system must be given unequivocal bus addresses. Only in this way can unequivocal addressing on the bus take place. As an option, the addresses can also be assigned via the bus.

The physical system settings are made using the master parameter set. This contains the master bus address and, for example, the baud rate, the timeout settings and the number of transmission repeats. Together with the master parameter set, a slave data set has to be stored for each slave to be activated. Each data set contains the parameterization and configuration data of the slave and the address vectors for the logical storage of the I/O data.

When the parameter sets are available, the master system begins to start up the slaves in succession on instruction by the user or automatically. The first diagnostic cycles indicate which slaves are present on the bus. Only those slaves which have sent a correct feedback message in the diagnostic cycle are then parameterised with the relevant data stored in the master in subsequent parameterization cycles. When this has been performed correctly, configuration cycles follow with the comparison of the specified configuration data in the master and the actual configuration data in the slaves. After the last diagnostic cycle, each slave which has not detected an error in the comparison is ready for operation. Each of these slaves is then automatically included in user data transfer by the master.

For diagnostic, the master maintains a diagnostic buffer for each slave, which can be read by the user. For simplified diagnostic, a group diagnostic field is maintained at the same time, showing bit by bit whether a slave has diagnostic data ready or not.

2.6 System behaviour

The system behaviour is mainly determined by the operating status of the class 1 DP master. Distinctions are made between four operating conditions:

- OFFLINE:
  Condition after the master system has been switched on. No data transfer through the bus.
- STOP:
  The master loads the bus parameters and initializes the diagnostic buffer. No data transfer takes place.
- CLEAR:
  The master parameterizes and configures the slaves via the bus. It reads the input data, but retains the output data in a secure condition.
- OPERATE:
  User data transfer is active. New output data are transmitted cyclically and the latest input data read.

The system behaviour on failure of a slave can be parameterized in the DP master, so that it switches over automatically from the OPERATE condition to the STOP condition. This also interrupts user data transfer to all other slaves and the module outputs are switched to the secure zero condition.

3 Configuration of the assembly

The standard requires every manufacturer of DP slaves and DP masters to com-pile equipment data sheets detailing the characteristics and capabilities of the DP devices. In addition, all these data are to be grouped together in a master file for the device and provided to the user.
The objective of this master device file is to facilitate uniform and simple configuration by means of input programs independent of particular manufacturers and based on stipulated structures and codes within the file.
The format of the master device file is clearly stipulated by the PROFIBUS-DP standard. It is maintained as an ASCII file and can therefore by generated or edited with any conventional text editor.

For the configuration of the other stations in the master assembly, the DP system configurator under Windows has to be used. This can load a specific device master file for each station. All the data required for the station parameter set are already present in that file and do not need to be entered manually. All the data loaded comprise the master device file for the system configurator in Access data base format, from which the stations can then be selected.
The stations are assigned in accordance with the actual bus structure and given bus addresses in the editor window of the configurator. The start addresses up to which the slave input datas are stored in and the output datas are taken from, must be forgiven for every slave station also.
Remark for CIF 30-DPM, COM-DPM and CIF 104.DPM user: the addresses are word offset addresses relative to the start value of the buffer. Modules which have uneven data lengths always occupy each started word.

4 Diagnostic functions

For diagnostic the diagnostic LEDs on the device and over the serial diagnostic interface the possibilities with the program ComPro can be used.

4.1 Diagnostic LEDs

To diagnostic the CIF 30-DPM the diagnostic LEDs on the front panel can be used when the device has finished its initialization-sequence. The upper yellow LED RDY shines continous after the operation system itself has initialized without an error and no hardware defect has determined.
The green LED RUN blinks cyclic, if the protocol tasks haven't determined nor an hardware error nor an parameterization error. In this case the tasks are ready to start the communication. The LED shines continous, if the device has a data exchange communication with at least one projected slave participant.
The LED ERR shines additionally, if a bus error occured. This can be for example possible, if an projected slave can't found on bus.
The yellow LED STA represent the hold-token of the master. As long as the device doesn't recognize any other master systems on the bus, the device retain the right to send. Therefore the LED shines continuous.

4.2 Trace functions

The trace buffer of the master serves to the storage of the intertask communication messages. The messages will be stored in this buffer to help in error debugging within the device.
Concerning this for each single communication task a trace filter can be activated with the program ComPro. For this choose the menu `diagnostic\filter...`. Herein the event and entry can be choosen with the space key. For the `event: always` and for the `entry: message` must be choosen to become the trace started. Recommendation: choose the trace of the USR_INTF-Task, because this task has the maintain communication activity.

To list the stored trace the menu `diagnostic\trace` must be activated. The intertask communication messages will be printed on screen. If the system has noticed an error the error number and the error text will also be shown. With the error text a direct location of the error will be possible and help to find the error source.

Before starting the trace print, a file destination can be selected where the trace contents will be written in. For this in the menu `diagnostic\protokol...` a filename must be inserted. If the output of the trace is finished the trace must be deactivated with `event:off` and `entry: message.`

4.3 Extended task state

With the program ComPro online variables of the tasks can be shown. Therefore the data base 'DPM' must be loaded after the ComPro startup screen.

With the menu point `online/task/status read..` the variables of the protocol tasks can be read out and shown online on the screen.

The menu point `status read..` contains four slection-possibilites:

variable	signification
all	global running states
PLC	PLC-Task system states
USR_INTF	USER_INTF-Task system states
FDL_ASPC	FDL_ASPC-Task system states

selection possibilities

4.3.1 Extended task state of the PLC-Task

State table 'variables':

variable	signification
PLC-task-version	specifies for which hardware the task was compiled
mode	specifies the active process data handshakemode

Possible entries in the version state field could be the identifier 'CIF' or the identifier 'COM'. The PLC-Task-version on the CIF 30-DPM, COM-DPM or CIF 104-DPM is compiled with the identifier 'CIF'. The identifier 'COM' is used at KPO 104-DPM.

The variable 'mode' has the possible entries 0, 1, 2, 3 or 4. The value represents the active process data delivery procedure between the host program and the board.

4.3.2 Extended task state of the USR_INTF-Task

State table 'Common variables':

variable	signification
Scheduler state	state value of the sceduler
Announced modules	number of the configured modules
Wrong parameters	number of faulty slave parameters data sets
Activated modules	number of the activated modules
Activated bus parameters	0= busparameters active, 255= busparameters inactive
Active data exchange	active Data_Exchange handle
Inactive data exchanges	number of modules to which no process data exchange is possible
Active queues	number stores jobs
Data Control Time	counter of the Data_Contol_Time
Min. interval-time	counter of the min.Slave_Intervals
Faulty station number	address of station detected as faulty
Timeout counter	supervision counter activated, when a bus short circuit is detected

Extended explanation of several status variables:

Basically the value of the 'Scheduler state' must be 13 or 14. This shows that the device works in the condition OPERATE. If the value is 5, the device works in the condition STOP. The reason for this can be the following:

- 1.) the device has activated the 'auto_clear' mode, because at least one slave could not be brought into the process data exchange during the supervision time. Please note the 'auto_clear' configuration in program SyConDP.
- 2.) The user program has set its condition to not ready, so the device stops the process data exchange to all stations. Please note the user specific parameters in program SyConDP.

The entry 'Wrong parameters' might not show any faulty data file. Othwerwise the status table 'parameters valid' must be examined, to get the faulty slave data set.
'activated bus parameters' must be 0 to show that the master bus parameters are active.
Especially important is the value of 'Inactive data exchanges'. Is shows globally the number of station which could not be brought into the process data exchange because of an error. So if the counter shows 0, every station could be activated without an error.
The 'Timeout counter' will be incremented, if no telegramms can be sent to the bus because of bus short circuits. If the counter contains a value different from 0, bus short circuits were detected during run time.

State table 'Running state'

variable	signification
Slave 0	Slave handler-state station address 0
...	...
Slave 126	Slave handler-state station address 126

For each projected slave the device has a status-value, which can be read out with the state table. The entries have following meaning:

slave handle state	meaning
0,2	slave is deactivated
3,4	connection establishing
5,6	parameterization phase
7,8,9,10	configuration phase
11,12	data transfer phase

The state for a slave runs through the listed values, until the data transfer phase is reached. If a state is stopped between 0 and 10, an error occured in the communication line of this slave:

state	signification
0,2	this slave is not projected, no communcation to the slave
3,4	the slave is not responding
5,6	parameter data are refused by the slave
7,8,9,10	no consensus between master and slave configration data

State table 'global status field'

variable	signification
Global state bits	global condition-bits
Master main state	00= Offline, 40= [Stop], 80= [Clear], C0= [Operate]
Error remote address	faulty bus address of 0-126= slave; 255= Master
Error event	exactly error value
Bus error counter	counter of detected heavy bus errors
Timeout counter	counter of bus time outs caused by bus short circuits
Slave 7-0 param.	slave 0-7 are parameterized
...	.
Slave 126-120 param.	slave 120-126 are parameterized
Slave 7-0 active	slave 0-7 in Data_Exchange-mode
...	.
Slave 126-120 active	slave 120-126 in Data_Exchange-mode
Slave 7-0 diagnostics	diagnostic bits for slave 7-0
...	.
Slave 126-120 diagnostics	diagnostic bits for Slave 126-120

The bit field 'Global state bits' serves to the collective-display of global error notifications, which can appear on the one hand at the device or on the other hand at the slave modules. Following meaning has assigned for 'Global state bits':

D7	D6	D5	D4	D3	D2	D1	D0
0	0	NRdy	Event	Fatal	NData	AClr	Ctrl

parameterization error

device braches in auto_clear_mode because of buserror

at least one slave is not communcating or notifies heavy mistake

because of heavy bus error, no further bus communication is possible

bus short circuits detected

user program is not ready

reserved

In the variable 'Error remote address' no value should have written down. It specify the bus address an error was noted. Is the value = 255 so the error within the master device was determined. 'Error event' supplements the error bus address in giving the appertaining error number.

Following error numbers are valid for 'Error event', if 'Error remote address' = 255:

err_event	signification	error source	help
0	no mistakes appear		
50	USR_INTF-Task not found	device	contact technical support
51	no global data-field	device	contact technical support
52	FDL-Task not found	device	contact technical support
53	PLC-Task not found	device	contact technical support
54	non existing master parameters	device	execute download of data base again
55	faulty parameter-value in the master parameters	project planning	contact technical support
56	non existing slave parameters	project planning	execute download of data base again
57	faulty parameter-value in a slave parameters datafile	project planning	contact technical support
58	double slave address	project planning	check projected addresses
59	projected send process data offset address of a participant outside the allowable border of 0- 255	project planning	check projected addresses

60	projected receive process data offset address of a participant outside the allowable border of 0- 255	project planning	check projected addresses
61	Data-areas of slaves are overlapping in the send process data	project planning	check projected addresses
62	Data-areas of slaves overlapping in the receive process data	project planning	check projected addresses
63	unknown process data handshake	warm start	check warm start parameters
64	free RAM exeeded	device	contact technical support
65	faulty slave parameter data sets	project planning	contact technical support
202	no segment for the treatment free	device	contact technical support
212	faulty reading of a data base	device	execute download of data base again
213	structure-surrender to operating system faulty	device	contact technical support

The following error numbers are valid for 'Err event', if 'Error remote address' is uneuqal to 255:

err_event	signification	error source	help
2	station reports overflow	master telegram	check length of configured slave configuration or parameter data.
3	request function of master is not activated in the station	master telegram	check slave if PROFIBUS-DP norm compatible
9	no answer-data, although the slave must reponse with data	slave	check configuration data of the station and compare it with the physical I/O data length
17	no response of the station	slave	check bus cable, check bus address of slave
18	master not into the logical token ring	device	check FDL-Address of master or highest-station-Address of other master systems. examine bus cableing to bus short circuits.

State table 'Communication error':

variable	signification
Slave 0	error number of slave address 0
...	...
Slave 126	error number of slave address 126

Basically no error value might have written down at none of the projected slaves. Otherwise the given error table is valid link for 'Error event' ('remote address' <> 255).

State table 'slave parameter set list':

variable	signification
Slave 0	0xff = parameter file valid, 0x00 = invalid, or not projected
...	...
Slave 126	0xff = parameter file valid, 0x00 = invalid, or not projected

As soon as a slave parameter file became valid, a 0xff is written down into the table. Not projected parameter files or invalid parameter files has written down the value of 0x00.

State table 'Disconnect report':

variable	signification
Slave 0	communication disconnect counter for slave 0
...	...
Slave 126	communication disconnect counter for slave 126

The state table shows the count of communication disconnects caused by bus errors of all handled stations. The counter will be inremented if additional request in case of errors have no effect to get reponse from the stations again. This will happen if the DP norm max retry limit is reached. See: SyConDP 'bus parameters' max retry limit).

State table 'Diagnostic report':

variable	signification
Slave 0	counter of diagnostic report of slave 0
...	...
Slave 126	counter of diagnostic report of slave 0

The state table shows the count of diagnostic reports of every slave station, which were reported deliberatly to the master. For every counted notification, the data exchange operation to this slave was left one DP cycle, to read out the prepared slave diagnostic data.

4.3.3 Extended task state of the FDL_ASPC-Task

State table 'Common variables':

variable	signification
Task state	task state
Last FDL error state	last FDL error state
Last FDL init error	initialization error of FDL
Last FDL runtime error	runtime error code
Last FDL message error	last message error number
ASPC2 bus short circuits	Count of 'bus syncon error' of the ASPC2
ASPC2 bus error	Count of 'bus error' of the ASPC2
Free application blocks	free application blocks of the software
Free SAP blocks	free SAP blocks of the software
Free CLASS2 blocks	free CLASS2 blocks of the software

While the system is running the condition 'task state' changes between the values of '10, 11, 21.' The FDL error values are changed only at runtime, if an error occurs between the main-processor and ASPC2-PROFIBUS-Controller. Is an error value displayed, please inform our hotline.

State table 'bus parameters':

variable		signification
Highest station address	HSA	highest station-address the master polls
Bus address of this master station	TS	the master bus address
Transmission rate		transmission rate, definition range of ASPC2
Maximum request retry in errorcase		retry-number of requestor
Slot time (Bit)	TSL	slot time
Transmitter fall time (Bit)	TQUI	transmitter fall time
Setup time (Bit)	TSET	setup time
Smallest station delay (Bit)	MIN-TSDR	minimum STATION-DELAY
Largest station delay (Bit)	MAX-TSDR	maximum STATION-DELAY
Target rotation time (Bit)	TTR	target rotation time
GAP update factor	G	gap update factor

The activated master parameters are shown in this table.

State table 'DDLM requests cl.1'

variable	signification
Set bus parameters	counter of 'set-bus-par' requests
Set slave address list	counter of 'set-slave-list' requests
Slave diagnostic request	counter of 'slave-diag' requests
Set parameters request	counter of 'set-prm' requests
Check configuration request	counter of 'check-cfg' requests
global control request	counter of 'global-control' requests
Data exchange request	counter of 'data-exchange' requests
Data exchange conf. pos.	counter of 'data-exchange' confirmation, positiv
Data exchange conf. neg.	counter of 'data-exchange' confirmation, negativ
Data exchange all request	counter of 'data-exchange-all' requests
Data exchange all conf. pos.	counter of 'data-exchange-all' confirmation, positiv
Data exchange all conf. neg.	counter of 'data-exchange-all' confirmation, negativ

The counter for master services of the PROFIBUS-DP are listed in this table. Basically the 'data exchange all' counter should increment at a faultless network only. The services 'set-slave-list', 'set-prm', 'chk-cfg', 'data-exchange' or 'slave-diag' might be incremented at faultless network for each projected slave only once. If an bus error occurs all these services are also increased.
If for example the service 'slave-diag' increases sporadic, maybe the bus cable has a defect or the conclusion-resistor in the cable is missing. Furthermore it can be possible that a slave reports an error in his diagnostic data, so that the master automatically starts the service 'slave_diag'.

State table 'DDLM requests cl.2'

variable	signification
Get configuration request	counter of 'Get_Cfg' requests
Get configuration conf. pos.	counter of 'Get_Cfg' confirmations positiv
Get configuration conf. neg.	counter of 'Get_Cfg' confirmations negativ
Read input request	counter of 'RD_Inp' requests
Read Input conf. pos.	counter of 'RD_Inp' confirmations positiv
Read Input conf. neg.	counter of 'RD_Inp' confirmations negativ
Read output request	counter of 'RD_Outp' requests
Read output conf. pos.	counter of 'RD_Outp' confirmations positiv
Read output conf. neg.	counter of 'RD_Outp' confirmations negativ
Set slave address request	counter of 'Set_Slave_Add' requests
Set slave address conf. pos.	counter of 'Set_Slave_Add' confirmations positiv
Set slave address conf. neg.	counter of 'Set_Slave_Add' confirmations negativ
Get master diag request	counter of 'Get_Master_Diag' requests
Get master diag conf. pos.	counter of 'Get_Master_Diag' confirmations positiv
Get master diag conf. neg.	counter of 'Get_Master_Diag' confirmations negativ
Get master diag indication	counter of 'Get_Master_Diag' indications
Get master diag response	counter of 'Get_Master_Diag' responses
Download request	counter of 'Download' requests
Download conf. pos.	counter of 'Download' confirmations positiv
Download conf. neg.	counter of 'Download' confirmations negativ
Download indication	counter of 'Download' indications
Download response	counter of 'Download' responses
Upload request	counter of 'Upload' requests
Upload conf. pos.	counter of 'Upload' confirmations positiv
Upload conf. neg.	counter of 'Upload' confirmations negativ
Upload indication	counter of 'Upload' indications
Upload response	counter of 'Upload' responses
Start sequence request	counter of 'Start_Seq' requests
Start sequence conf. pos.	counter of 'Start_Seq' confirmations positiv
Start sequence conf. neg.	counter of 'Start_Seq' confirmations negativ
Start sequence indication	counter of 'Start_Seq' indications
Start sequence response	counter of 'Start_Seq' responses
End sequence request	counter of 'End_Seq' requests
End sequence conf. pos.	counter of 'End_Seq' confirmations positiv
End sequence conf. neg.	counter of 'End_Seq' confirmations negativ
End sequence indication	counter of 'End_Seq' indications
End sequence response	counter of 'End_Seq' responses
Act.param. brct. request	counter of 'Act_Param_Brct' requests
Act.param. brct. conf. pos.	counter of 'Act_Param_Brct' confirmations positiv
Act.param. brct. conf. neg.	counter of 'Act_Param_Brct' confirmations negative
Act.param brct. indication	counter of 'Act_Param_Brct' indications
Act.param. brct. response	counter of 'Act_Param_Brct' responses
Act.param. request	counter of 'Act_Param' requests
Act.param. conf. pos.	counter of 'Act_Param' confirmations positiv

Act.param. conf. neg.	counter of 'Act_Param' confirmations negativ
Act.param. indication	counter of 'Act_Param' indications
Act.param. response	counter of 'Act_Param' responses

The counter for Class 2 services of the PROFIBUS-DP are listed in this table. Here, request (out going) and indications (in coming) are counted.

Not all functions are supported in this implementation.

State table 'FDL-requests'

variable	signification
SDA Request	counter of 'SDA' requests
SDA Confirmation pos.	counter of 'SDA' confirmations, positiv
SDA Confirmation neg.	counter of 'SDA' confirmations, negativ
SDA Indication	counter of 'SDA' indications
SDN Request	counter of 'SDN' requests
SDN Confirmation pos.	counter of 'SDN' confirmations, positiv
SDN Confirmation neg.	counter of 'SDN' confirmations, negativ
SDN Indication	counter of 'SDN' indications
SRD Request	counter of 'SRD' requests
SRD Confirmation pos.	counter of 'SRD' confirmations, positiv
SRD Confirmation neg.	counter of 'SRD' confirmations, negativ
SRD Indication	counter of 'SRD' indications
SRD-Update Request	counter of 'SRD' update requests
SRD-Update Con. pos.	counter of 'SRD' update confirmations, positiv
SRD-Update Con. neg.	counter of 'SRD' update confirmations, negativ

The counter for FMS services of the PROFIBUS are listed in this table. Here, requests (out going), its confirmation (positiv or negativ) and indications (in coming) are counted.

Not all functions are supported in this implementation.

State table 'FMA requests'

variable	signification
SAP act. request	counter of 'SAP Aktivate' requests
SAP act. confirmation pos.	counter of 'SAP Aktivate' confirmations, positiv
SAP act. confirmation neg.	counter of 'SAP Aktivate' confirmations, negativ
RSAP act. request	counter of 'RSAP Aktivate' requests
RSAP act. confirmation pos.	counter of 'RSAP Aktivate' confirmations, positiv
RSAP act. confirmation neg.	counter of 'RSAP Aktivate' confirmations, negativ
SAP deact. request	counter of 'SAP Deaktivate' requests
SAP deact. confirmation pos.	counter of 'SAP Deaktivate' confirmations, positiv
SAP deact. confirmation neg.	counter of 'SAP Deaktivate' confirmations, negativ
LiveList request	counter of 'LiveList' requests
LiveList confirmation pos.	counter of 'LiveList' confirmations, positiv
LiveList confirmation neg.	counter of 'LiveList' confirmations, negativ

The counter for FMA services of the PROFIBUS are listed in this table. Here, requests (out going), its confirmation (positiv or negativ) and indications (in coming) are counted.

Not all functions are supported in this implementation.

State table 'Data exchange retry'

variable	signification
Slave 0	counter of retry for slave with bus address 0
Slave 1	counter of retry for slave with bus address 1
Slave 2	counter of retry for slave with bus address 2
.....	....
Slave 126	counter of retry for slave with bus address 126

If a telegram for a slave had to be send again 1 or up to 7 times, this will be counted in this structure for every slave. The count of retries itself per telegram is not counted and can not be read out.

State table 'DP: activated slave'

variable	signification
Slave 0	Inactive(=0)/active(=1) slave with bus address 0
Slave 1	Inactive(=0)/active(=1) slave with bus address 1
Slave 2	Inactive(=0)/active(=1) slave with bus address 2
.....	....
Slave 126	Inactive(=0)/active(=1) slave with bus address 126

This structure displays information about the slave that are activated from this master. Value 0 means slave is not activated, value 1 means slave activated.

5 Handing over the process datas

The access of the user program and the device on the process data must be sychronized to avoid data inconsistency of interralated data. Therefore the device supports several access handshakes just to handle the most different requests of the user.

For the handing over the process data between the user and the device the following operating modes can be used:

synchronization	controlling	data consistency	supporting
bus sychronous	device controlled	observed	master/slave
bus asychronous (buffered)	device controlled	observed	master/slave
none	uncontrolled	not observed	master/slave
bus asychronous (buffered)	host controlled	observed	master/slave
bus sychronous	host controlled	observed	only master

survey of the operating modes

Because of the different function manners of the delivery procedures, it must be guaranteed that the user-program or the used software-driver works in the same operation mode as the chosen operating mode of the device.
Is no user-program available or no software-driver in use, the device can be parameterized in operating mode 'standard, uncontrolled', because only in this mode the delivery of process data requires no sychronization mechanism. In this case the user can grab in to the send and receive process data permanently. CAUTION! However only a byte-consistency of the process-data is guaranteed in this mode.

6 The example application

For the online viewing and changing of the process data an example program can be loaded in the system-package. Under MS-DOS or compatible MS-DOS-window or under Windows the batch file 'IOVIEW.BAT' can be called. Is starts the DOS example application 'IOVIEW.EXE'. The batch file includes the start address of the device as delivery parameter like '/A:CA00'. Be sure that the device works in 'standard, uncontrolled' operating mode to establish the communication between the program and the device, because the program supports no sychronization. After its start the program checks, if the communication with the device is possible. When it recognizes the card, the card will be initialized. Afterwards the send and receive process data contents are displayed in word-size. With the 'cursor up' or 'cursor down' key it is possible to change between the 256 different send process datawords and with the 'cursor left' and 'cursor right' key the receive process data words can be selected. With the 'i' or the 'd' key the contents of the shown send process data can be increased or decreased. Addition to the process data the three bases states of the card are displayed. 'RDY' reports that the device has finished its initialization without an error. 'RUN' is reported, if non of the implemented software task has recognized an error. The condition 'COM' is activated, if at least one of the projected slaves reaches the communication state 'data_exchange' and data between the device and the slave is transferred.
The <*ESC*>-button quit the the program.

7 Technical Data

PROFIBUS-DP Master

- Combi Master (PROFIBUS-DP Master/PROFIBUS-FMS Master) devices:
  - CIF 30-PB, CIF 50-PB, CIF 60-PB, CIF 104-PB, COM-PB
  - PKV 20-PB uses COM-PB
- PROFIBUS-DP Master devices:
  - CIF 30-DPM, COM-DPM
  - PKV 20-DPM uses COM-DPM
  - KPO 104-DPM

ASIC: ASPC2

Max. 125 Slaves

Max. 244 Bytes per Slave

DPV1 extension available on: CIF xx-PB and COM-PB

I/O Data

- On Combi Master devices: CIF xx-PB, COM-PB, PKV 20-PB
  - 7 KByte: 3584 Bytes for Input, 3584 Bytes for Output
- On PROFIBUS-DP Master devices: CIF xx-DPM, COM-DPM, PKV 20-DPM, KPO 104-DPM
  - 1 KByte: 512 Bytes for Input, 512 Bytes for Output

Transmission rate: 9,6 kBaud - 12 MBaud

9,6 kBaud, 19,2 kBaud, 93,75 kBaud, 187,5 kBaud, 500 kBaud,

1,5 MBaud, 3 MBaud, 6 MBaud, 12 MBaud