FBG10

Field Bus Gateway

User Manual

JVL Industri Elektronik A/S

LB0046-02-GB

Revised 19.11.2002

Contents

JVL Industri Elektronik A/S reserves the right to make changes to information contained in this manual without prior notice.
Similarly JVL Industri Elektronik A/S assumes no liability for printing errors or other omissions or discrepancies in this user manual.

MotoWare is a registered trademark of JVL Industri Elektronik A/S

JVL Industri Elektronik A/S
Blokken 42
DK-3460 Birkerød
Denmark
Tlf. +45 45 82 44 40
Fax. +45 45 82 55 50
e-mail: jvl@jvl.dk
Internet: http://www.jvl.dk

1 Introduction

The FBG series of gateways is used to interface different fieldbus systems, such as Profibus, Interbus etc., to JVL units equipped with an RS232 interface. They thus function as gateways between a fieldbus and the RS232, translating commands bidirectionally between the two systems.

In this way it is possible for example to use a PLC to control velocities, displacement and register contents directly from a fieldbus, and to read error codes, position, register contents, inputs and outputs from connected JVL modules. Important operating parameters, I/O status, etc., are continuously monitored by the FBG10 and can therefore be quickly read by the fieldbus. The FBG10 series of Field Bus Gateways thus provides an inexpensive solution for applications that use a fieldbus together with JVL modules which only have an RS232 interface.

Customised models of the FBG10 can also be produced for RS232 protocols other than JVL´s own.

- Fieldbus protocols:
  Profibus-DP slave
  CAN - OPEN*
  Interbus-S slave*
  P-Net slave*
  DeviceNet slave*
  Other protocols on request
- RS232 interface
- Fieldbus slave address set on dip-switch
- DIN rail mounting
- 10-30 VDC operation
- Compact unit which is easily mounted
- LED indication for Power, Ready, Busy RS232/Fieldbus
- Differential bus interfaces with opto-couplers
- DSUB and Combicon connectors
- CE approved
- Dual port RAM for exchange of data
- Can be connected to JVL Controllers AMC1x, AMC2x,SMC2x, DMC10 and Indexer SMI3x

* under development (01-10-99)

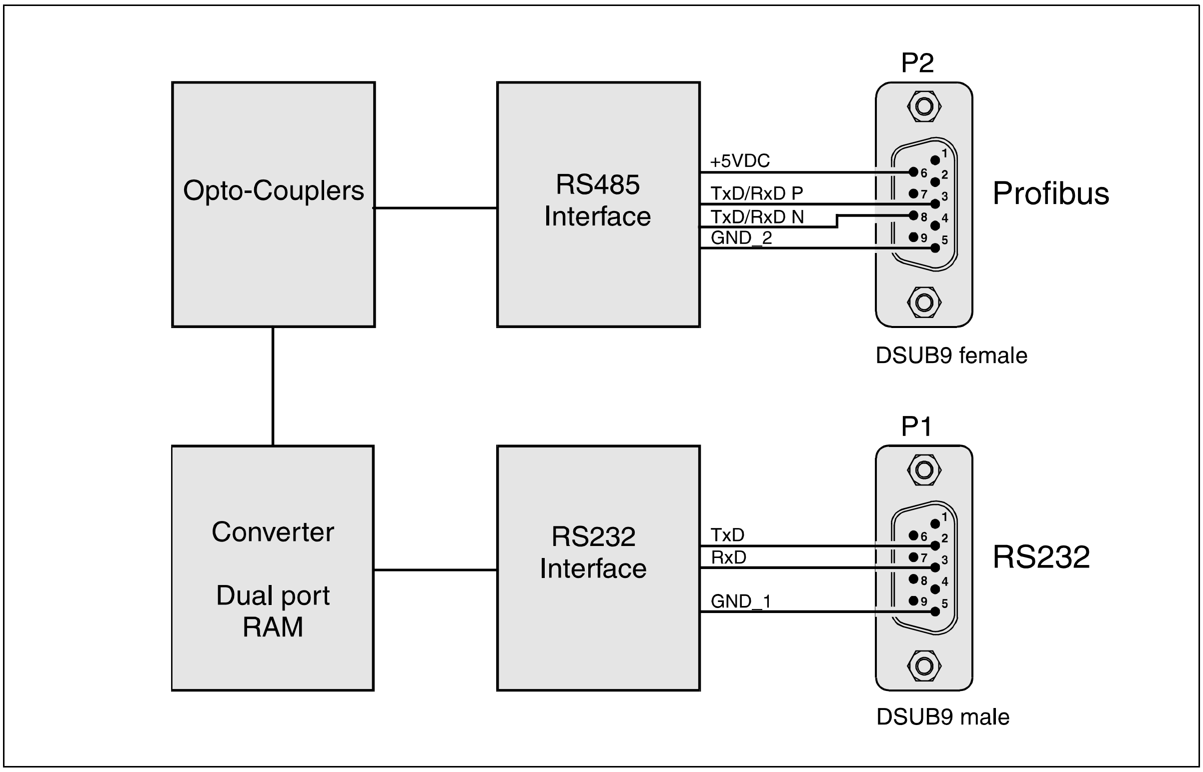
1.1 Field Bus Gateway FBG-10

FBG10 is a PROFIBUS-DP / RS 232 - converter, which makes it possible to communicate via a PROFIBUS with the following JVL-motor controllers:

SMI30/31 Step- and Servo Motor Indexers
SMC35 Step Motor Controllers
DMC10 DC- Servo Motor Controller
AMC10 AC- Servo Motor Controller up to 1 kW
AMC20 AC- Servo Motor Controller up to 3 kW

Direct communication with these controllers takes place via an ASCII-based RS 232 protocol. FBG10 converts commands from PROFIBUS-commands to ASCII-commands, and converts ASCII-response to PROFIBUS-response.

Reference is made to DIN E 19245 part 3 for definition of PROFIBUS-DP.

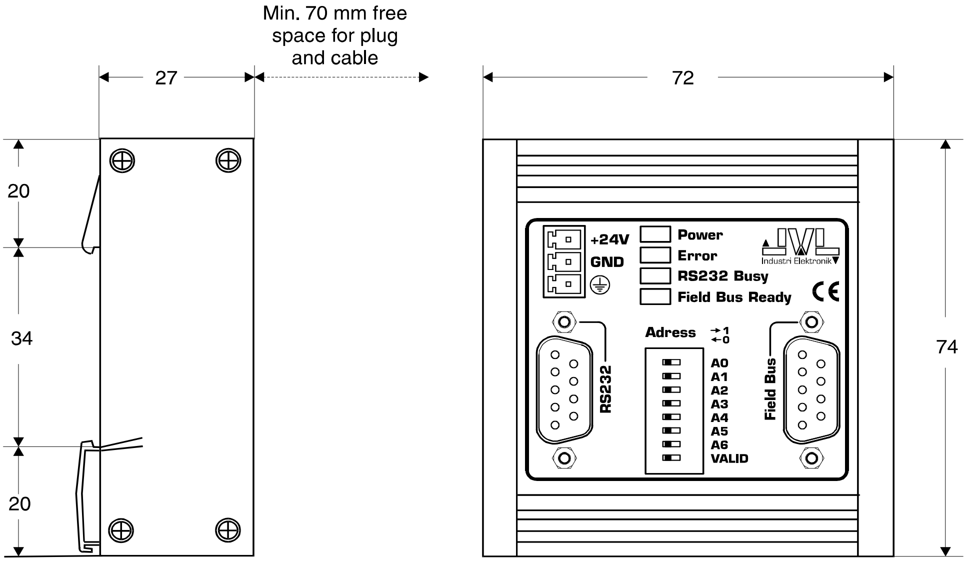


2 Installation

2.1 Introduction

FBG10 is designed for DIN-rail mounting in a control panel.

Drawing 2.0.1



Fits to "DIN" rail:
DIN VDE 0611 part3, DIN EN 50022
EN 50035, EN 50045

All dimensions in mm

2.2 Connections, PROFIBUS

For connection to PROFIBUS-DP, a 9-pole SUBD-connector, (female), designated "Field bus" is available on the front panel.

Connections:

	Pin No.	**Description**
RxD/TxD-P:	3	
DGND:	5	Ground
P5V	6	Supply Voltage
RxD/TxD.N:	8	

2.3 Connections, RS232

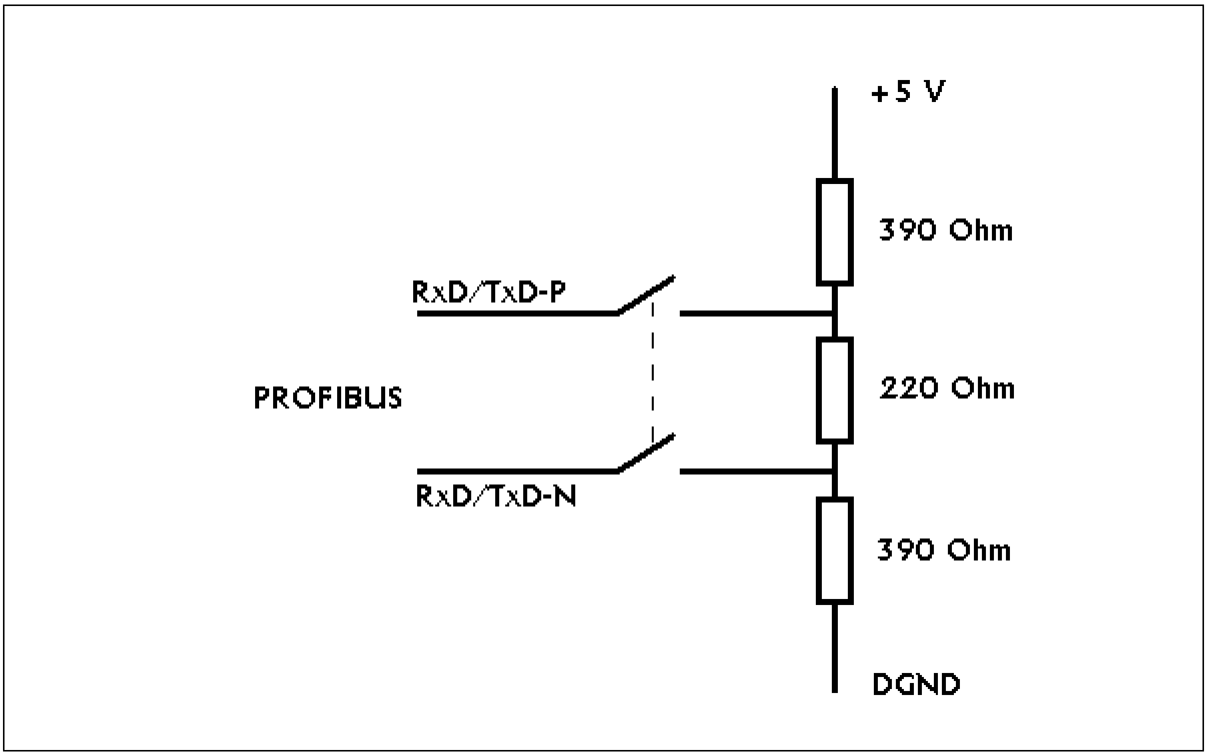
For connection to an RS 232 interface, a 9-pole SUBD-connector, (male), designated "RS232" is available on the front panel.

Connections:

	Pin No.	Description
Rx:	2	Receive from controller
Tx::	3	Transmit to controller
GND:	5	RS232 interface ground

2.4 Bus Termination, PROFIBUS

DIN E 19245 requires that PROFIBUS is terminated at both ends with a resistor network as shown below:



FBG10 does *not* include this network and connection to PROFIBUS should therefore take place by means of a SUBD-connector with an integrated resistor network. This is however only necessary if the FBG10-unit is physically connected as the last unit on PROFIBUS.

Siemens and other suppliers can deliver connectors with built-in resistors and a switch to activate/deactivate the resistors. They are available in 35 degree and 180 degree versions.

Siemens connectors:

35 degree cable outlet	180 degree cable outlet (Straight)
Siemens no. 6ES7972-0BA40-0XA0 LP-no. 0489220634 Solar-no. 2686050045 EAN: 4025515062929	Siemens no. 6GK1500-0FC00 LP-no. 0489211892 Solar-no. 2686050184 EAN: 5703943006461

2.5 Setting of Unit address

To set the unit address, a DIP-switch is available on the front panel of the FBG10

All of the unit addresses from 0 ... 127 can be set by means of the switches A6 ... A0 . The setting of the switch designated "VALID" determines if the address set is valid.

Function:
— FBG10 contains an EEPROM in which the last set unit address is stored.
— DIP-switches are read at each "power on".
— If VALID = 1, then FBG10 will get the address which A6 ... A0 indicates and this address is at the same time stored in EEPROM.
— If VALID = 0, then FBG10 will get the address which is stored in EEPROM.
—
— Setting of the unit address is therefore done as follows:

1. Set A6 ... A0 to the required address.
2. Set VALID = 1.
3. Connect supply voltage and wait 1 sec.
4. Set VALID = 0.

The set address is now stored in EEPROM, and will be used at subsequent "power on".

2.6 LED Functions

FBG10 contains 4 LEDs on the front panel for indication of status.

POWER:
Indicates that the supply voltage is connected. As the diode is activated via the program, the diode furthermore may indicate that the program is executed correctly in the internal controller.

ERROR:
Indicates that communication on PROFIBUS or RS 232 is in error. The PROFIBUS-master may for example have sent an unknown command or FBG10 has not been able to convert a value from the motor controller. The PROFIBUS-master can read the exact cause of the error. See section *3.7.1, "Status description"*, page 24.

FIELD BUS READY:
Indicates that the FBG10 is active concerning communication on PROFIBUS. This implies that the physical connection is in order, and that the master on PROFIBUS has accepted the FBG10 unit.

RS 232 BUSY:
This diode is lit as long as communication between a motor controller and the FBG10 via RS 232 takes place.

3 Interface for PROFIBUS-DP

3.1 Introduction

PROFIBUS-DP is suited for fast communication between a master and slaves on PROFIBUS.

PROFIBUS-master sends cyclical process data, (output data), to each slave on PROFIBUS, and receives following process data, (input data), from the slave.

3.2 Configuration

In order to achieve fast communication, both master and slaves on PROFIBUS are acquainted with the output- and input-data format. These formats are determined under configuration of each slave.

Data format for the FBG10 is configured by:

8 byte output.
8 byte input.

See section *3.6, "Description of Output Data"*, page 21 and section *3.7, "Description of Input Data"*, page 24.

3.3 Ident-number

Each DP-slave is allocated an Ident-number, which identifies the slave to the master.

The Ident-number for the FBG10 is:

```
050C HEX
```

3.4 Set Station Address

Certain DP-slaves support a change of unit address via PROFIBUS. This function is not supported by the FBG10. Instead the unit address is set manually via DIP-switches. See section *2.5, "Setting of Unit address"*, page 12.

3.5 Sync and Freeze mode

Certain DP-slaves support the functions SYNC and FREEZE:

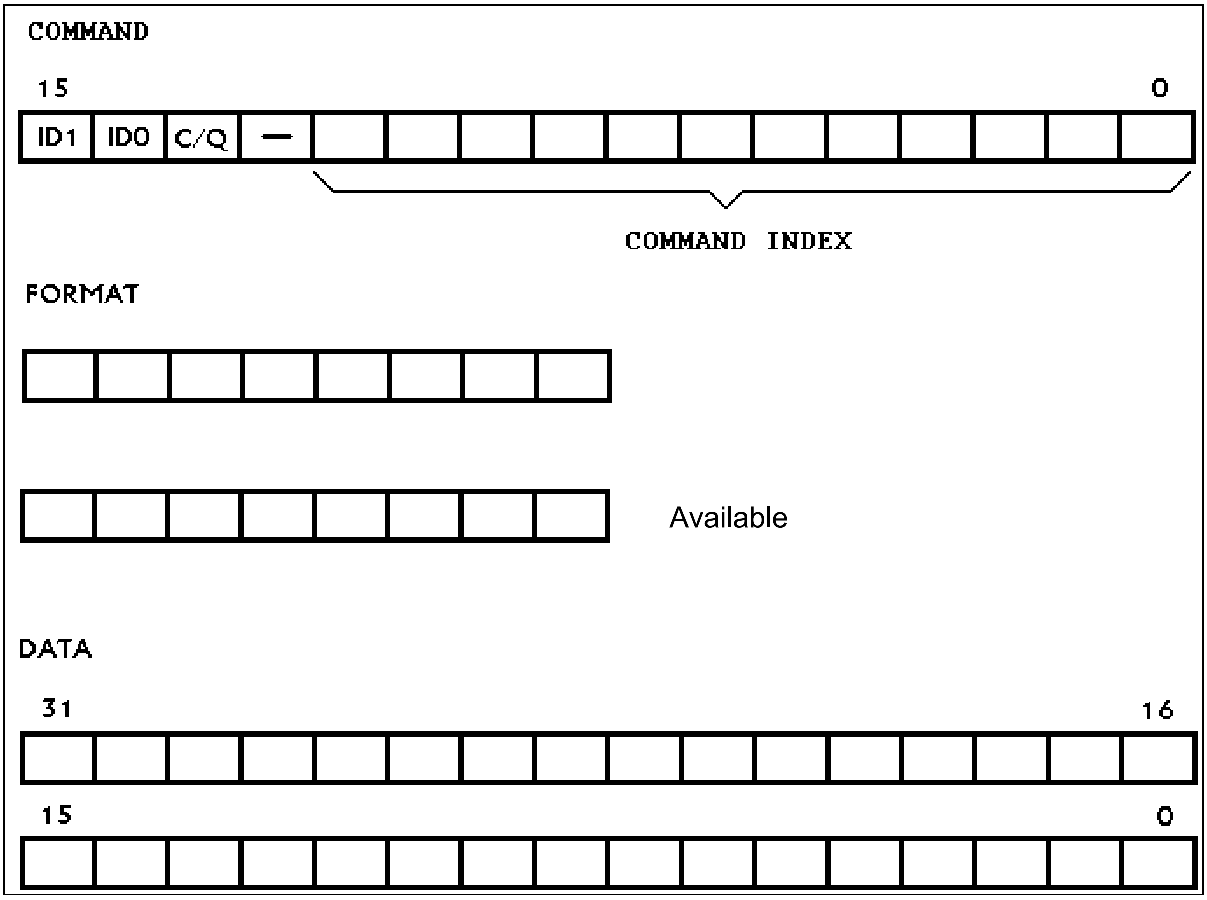
SYNC:
This function enables that process data can be sent to a group of slaves. These data however only have effect when a SYNC-command is sent to the group. As the whole group receives this command synchronously, the new process data are also executed synchronously even if data is received asynchronously.

FREEZE:
This function causes a group of DP-slaves to freeze data from the process. After the master has given this command, the master can asynchronously fetch data from each slave in the group. These data will have been read synchronously by the selected group of DP-slaves.

As the RS 232 protocol between the FBG10 and a motor controller does not include corresponding synchronisation-mechanisms, these functions are not supported by the FBG10 on PROFIBUS.

3.6 Description of Output Data

Output data, (data from master to DP-slave), is configured to 8 byte:



3.6.1 Command Description

BYTE 1+2 in output data form a word, (bit 15 ... 0), where the individual bits have the following meaning:

Bit 15+14:
The bits ID1,ID0 are set equal to the value of a two-bit counter in the master, which counts in the sequence: 00, 01, 10, 11, 00, etc. Every time the master sends new process data to an FBG10-unit, this counter value is incremented by the master. The FBG10 contains a corresponding counter. When the value of the received ID1,ID0 is equal to the counter value + 1 of FBG10, FBG10 will make the conversion between PROFIBUS and RS 232, and will thereafter increment the counter value of FBG10. FBG10's counter value can be read by the master in input data, (see section *3.7*). When a command is executed, the content of the master- and FBG10- counters will be equal. Thus the FBG10 can determine if new data processing must take place from the value of ID1,ID0 compared with its own counter value. Similarly the master can determine if a command has been executed from the counter value of FBG10.

Bit 13:
The bit C/Q, (Command/Question), indicates to the FBG10 if the master wants a command executed, (C/Q = 1), or if the master sends a request to FBG10, (C/Q = 0). If it is a command, FBG10 will write a new value to a register in the motor controller. If it is a request, FBG10 will read a value from a register in the motor controller and return the value to the master.

Bit 12:
Available.

Bit 11...0:
These bits forms the index for the Read/Write register in the motor controller. Which index each register is allocated to can be seen from the conversion table; see section *5, "Conversion Table"*, page 35.

3.6.2 Format description

FBG10 supports a series of different data formats on PROFIBUS, as well as register values on RS 232 can be presented in different ASCII-formats.

BYTE 3 in the output data indicates which format has been used in connection with a command, or which format must be used by a request.

The following formats are supported:

Format=0:
Default format / no data. Used in connection with a command, this designation means that no data belongs to the command. Used in connection with a request, this designation means that default format must be used. The conversion table, (see section *5, "Conversion Table"*, page 35), allocates a default format to each of the motor controller´s register values. On request, the format indication of the table will be used, when the register value is transmitted to the master on PROFIBUS.

Format=1:
Unsigned integer, (16 bit). See section *4.3.1, "Unsigned integer, (word)"*, page 30

Format=2:
Signed integer, (16 bit). See section *4.3.2, "Signed integer"*, page 30.

Format=3:
Long integer, (32 bit). See section *4.3.3, "Long integer"*, page 30.

Format=4:
Floating point, (32 bit IEEE). See section *4.3.4, "Floating point"*, page 30.

Format=5:
Binary, (16 bit). See section *4.3.5, "Binary"*, page 31.

Format=6:
Fixed point * 1.0, (32 bit). See section *4.3.6, "Fixed point * 1.0"*, page 31.

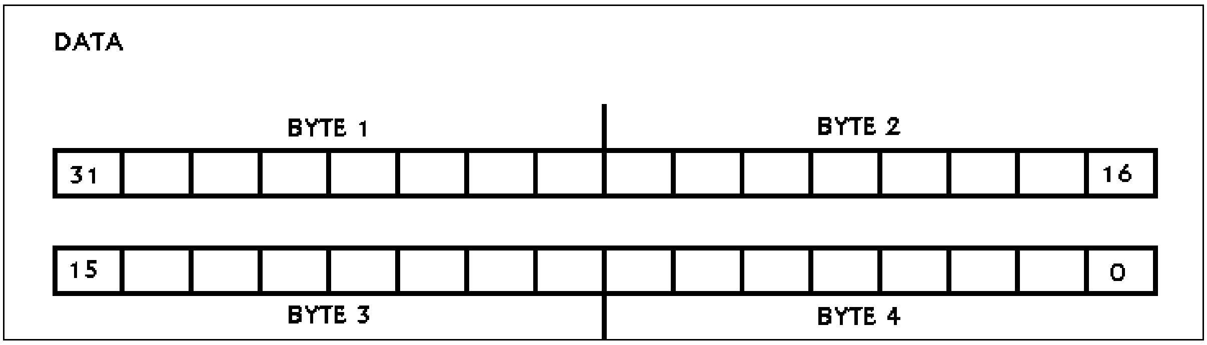
Format=7:
Fixed point * 10.0, (32 bit). See section *4.3.7, "Fixed point * 10.0"*, page 32.

Format=8:
Fixed point * 100.0, (32 bit). See section *4.3.8, "Fixed point * 100.0"*, page 32.

Format=9:
Fixed point * 1000.0, (32 bit). See section *4.3.9, "Fixed point * 1000.0"*, page 32.

3.6.3 Data description

On PROFIBUS, data can be represented by up to 4 bytes, (32 bit).



Note:

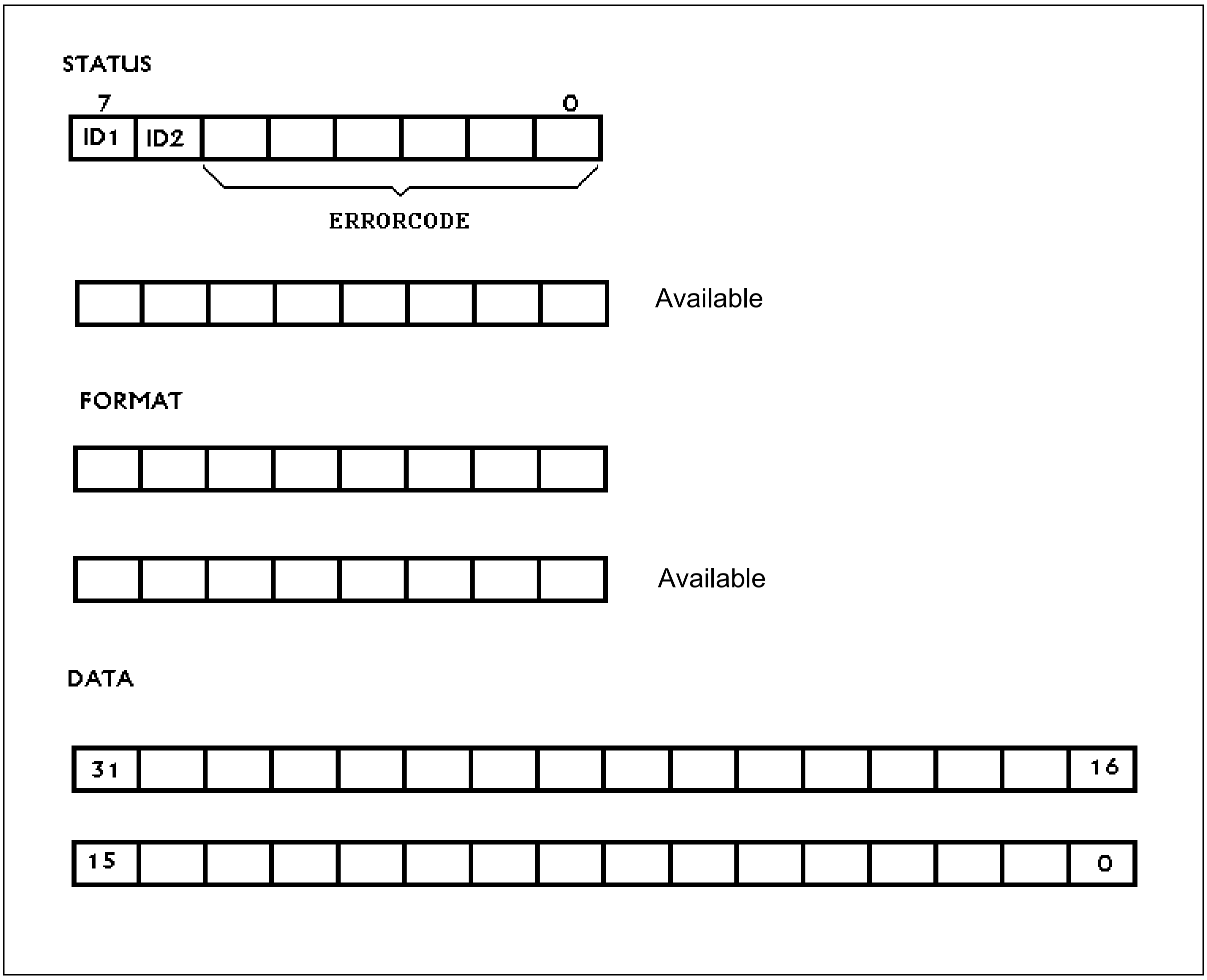
— In 16-bit formats the value is given in byte 3 and byte 4, with byte 3 as the most significant.

— In 32-bit formats the value is given in byte 1 ... byte 4, with byte 1 as the most significant.

See section *4.3.1, "Unsigned integer, (word)"*, page 30 to section *4.3.9, "Fixed point * 1000.0"*, page 32 for more detailed descriptions.

3.7 Description of Input Data

Input data, (data from DP-slave to master), is configured to 8 byte:



3.7.1 Status description

Status is read as a byte, in which each bit has the following meaning:

Bit 7...6:
The bits ID1,ID0 are set equal to the value of a two-bit counter in FBG10, which counts in the sequence: 00, 01, 10, 11, 00, etc. Each time the FBG10 has executed a command, this counter value is incremented. From the FBG10 counter value, the master can determine whether a command is executed.

Bit 5...0:
Each time a command is executed by the FBG10, the master can read an error code from these bits. The error code has the following meaning:

00:	No error.
01:	The ID1, ID0 bits in last command followed in correct sequence.
02:	Unknown command index in last command. See legal indexes in conversion table in section *5, "Conversion Table"*, page 35.
03:	Unknown format in last command.
04:	Data conversion error.
05:	Motor controller has not reacted on command within expected time.

06: Error in response from motor controller.
07: Other error: Master should read register ES for diagnosis.

Re error code 01: The FBG10 is in this condition if an ID1, ID0 received from PROFIBUS has not followed the correct sequence. The Error lamp will be lit in this condition. The condition is reset by the next command with correct sequence.

Re error code 02: The FBG10 is in this condition if last command is unknown. The condition is reset by the next recognised command.

Re error code 05: The FBG10 is in this condition if the controller has not transmitted to PROFI232 within the expected time. The condition is reset by the next command.

3.7.2 Format description

BYTE 3 in input data indicates which format has been used in connection with a request.

The following formats are supported:

Format=1:
Unsigned integer, (16 bit). See section *4.3.1, "Unsigned integer, (word)"*, page 30

Format=2:
Signed integer, (16 bit). See section *4.3.2, "Signed integer"*, page 30.

Format=3:
Long integer, (32 bit). See section *4.3.3, "Long integer"*, page 30.

Format=4:
Floating point, (32 bit IEEE). See section *4.3.4, "Floating point"*, page 30.

Format=5:
Binary, (16 bit). See section *4.3.5, "Binary"*, page 31.

Format=6:
Fixed point * 1.0, (32 bit). See section *4.3.6, "Fixed point * 1.0"*, page 31.

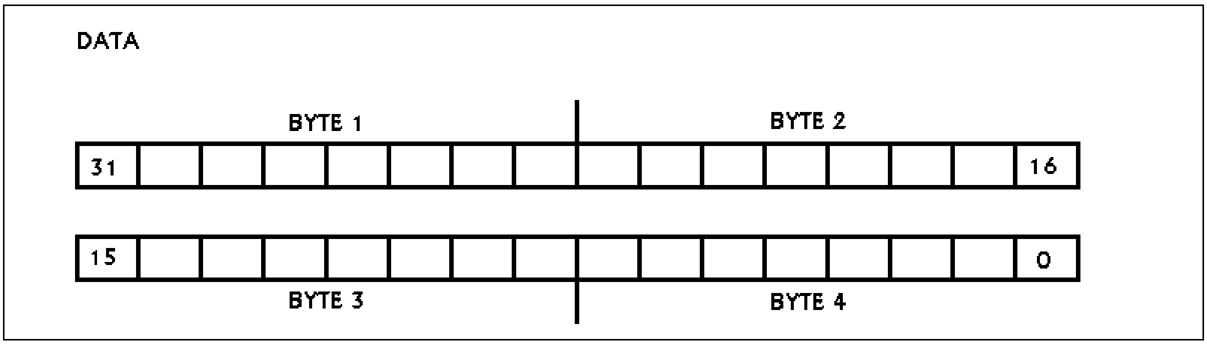
Format=7:
Fixed point * 10.0, (32 bit). See section *4.3.7, "Fixed point * 10.0"*, page 32.

Format=8:
Fixed point * 100.0, (32 bit). See section *4.3.8, "Fixed point * 100.0"*, page 32.

Format=9:
Fixed point * 1000.0, (32 bit). See section *4.3.9, "Fixed point * 1000.0"*, page 32.

3.7.3 Data description

Data on PROFIBUS can be represented by up to 4 bytes, (32 bit).



Note:

- In 16-bit formats the value is given in byte 3 and byte 4, with byte3 as the most significant.
- In 32-bit formats the value is given in byte 1 ... byte 4, with byte 1 as the most significant.

See section *4.3.1*, page 30 to section *4.3.9*, page 32 for more detailed descriptions.

4 Interface for RS232

4.1 Introduction

The protocol for RS 232 communication to a motor controller is ASCII-based.

A Motor controller will normally only transmit to the FBG10 in response to a command/request sent from FBG10. In connection with error conditions in the motor controller, the controller can however send error messages unprovoked. These unprovoked messages are not supported by the FBG10, but will result in a conversion error. See section *3.7.1, "Status description"*, page 24.

Certain commands that are sent to a controller will result in lengthy statements from the motor controller. For example the command: "Show servo constants", (K), results in the response:

```
KD=70
KI=10
KP=40
KVFF=10
IL=100
BIAS=0
```

As the FBG10 cannot convert this to a PROFIBUS response, such commands are not supported. Supported commands can be seen from the conversion table in section *5, "Conversion Table"*, page 35

4.2 Command Description

The syntax for a command to a motor controller can be written as follows:

```
Command        :=  command> [ <value> ] <cr>

<command>      :=  'AC' | 'BIAS' | 'CA' | ..... | 'ZS'

<value>        :=   '=' [<sign>] <digits> [ '.' [ <digits> ] ] <cr>

<sign>         :=  '+' | '-'

<digits>       := '0' .. '9' [ <digits> ]

<cr>           := char( 13 )
```

Commands can be divided in three groups:

1. Commands which assign a new value to a register in the motor controller.
2. Commands which perform control of a motor controller, for example 'RESET'.
3. Commands which request contents of a register or about a condition.

Re 1. These commands always include <value>. The PROFIBUS master must set the bit C/Q = 1 as well as Format > 0 in sending this type of command. See section *3.6.1, "Command Description"*, page 21 and section *3.6.2, "Format description"*, page 22. Example:

Command	Response
AC=123 <cr>	'Y' <cr>

Re 2. These commands omit <value>. The master on PROFIBUS must set the bit C/Q = 1 and set Format = 0 in sending this type of command. See section *3.6.1, "Command Description"*, page 21 and section *3.6.2, "Format description"*, page 22. Example:

Command	Response
SH <cr>	'Y' <cr>

Re 3. These commands omit <value>. The master on PROFIBUS must set the bit C/Q = 0 and set Format = 0 ... 9 in sending this type of command. See section *3.6.1, "Command Description"*, page 21 and section *3.6.2, "Format description"*, page 22. Example:

Command	Response
AC <cr>	AC=123 <cr>

4.3 Data-format description

As mentioned in section *3.6.2, "Format description"*, page 22 the FBG10 supports a series of formats. These are described in section *4.3.1* to section *4.3.9*.

4.3.1 Unsigned integer, (word)

Format = 1

PROFIBUS format:
Data format:16 bit
Number range: 0 ... 0xFFFF

RS 232 format:
Data format: ASCII
Number range: '0' ... '65535'

4.3.2 Signed integer

Format = 2.

PROFIBUS format:
Data format: 16 bit
Number range: 0x8000 ... 0x7FFF

RS 232 format:
Data format: ASCII
Number range: '-32768' ... '+32767'

4.3.3 Long integer

Format = 3.

PROFIBUS format:
Data format: 32 bit
Number range: 0x80000000 ... 0x7FFFFFFF

RS 232 format:
Data format: ASCII
Number range: '-2147483648' ... '+2147483647'

4.3.4 Floating point

Format = 4

PROFIBUS format:
Data format: 32 bit
Number range: Defined in IEEE

RS 232 format:
Data format: See below.
Number range: Defined in IEEE

Floating point :=
[<sign>] <digits> ['.' [<digits>]] 'E' [<sign>] <digits>

<sign> := '+' | '-'

<digits> := '0' ... '9' [<digits>]

Example:
Floating point := '23.45E-12'

Note: As can be seen in section *4.2, "Command Description"*, page 29, JVL motor controllers do not support exponential notation in connection with real numbers. Format=4 can therefore not be used in connection with JVL motor controllers. Use instead a suitable fixed point format.

See section *4.3.6* to section *4.3.9*.

4.3.5 Binary

Format = 5.

PROFIBUS format:
Data format: 16 bit
Number range: 0 ... 0xFFFF

RS 232 format:
Data format: ASCII
Number range: '0000000000000000' ... '1111111111111111'

4.3.6 Fixed point * 1.0

Format = 6.

PROFIBUS format:
Data format: 32 bit
Number range: 0x80000000 ... 0x7FFFFFFF

RS 232 format:
Data format: ASCII
Number range: '-2147483648.0' ... '+2147483647.0'

In this fixed point format, the least significant bit in PROFIBUS-data has the value: 1.0.

Example:

PROFIBUS-data	RS 232 data
0x401	'1025.0'

4.3.7 Fixed point * 10.0

Format = 7.

PROFIBUS format:
Data format: 32 bit
Number range: 0x80000000 ... 0x7FFFFFFF

RS 232 format:
Data format: ASCII
Number range: '-214748364.8' ... '+214748364.7'

In this fixed point format, the least significant bit in PROFIBUS-data has the value: 0.1

Example:

PROFIBUS-data	RS 232 data
0x401	'102.5'

4.3.8 Fixed point * 100.0

Format = 7.

PROFIBUS format:
Data format: 32 bit
Number range: 0x80000000 ... 0x7FFFFFFF

RS 232 format:
Data format: ASCII
Number range: '-21474836.48' ... '+21474836.47'

In this fixed point format, the least significant bit in PROFIBUS-data has the value: 0.01

Example:

PROFIBUS-data	RS 232 data
0x401	'10.25'

4.3.9 Fixed point * 1000.0

Format = 7.

PROFIBUS format:
Data format: 32 bit
Number range: 0x80000000 ... 0x7FFFFFFF

RS 232 format:
Data format: ASCII
Number range: '-2147483.648' ... '+2147483.647'

Example:

PROFIBUS-data	RS 232 data
0x401	'1.025'

5 Conversion Table

5.1 Conversion Table

The FBG10 contains a conversion table which is shown below.

This conversion indicates:

INDEX:
The value which the master writes as output data in COMMAND. See section *3.6, "Description of Output Data"*, page 21.

COMMAND NAME:
The ASCII-string which the command is converted to on the RS 232 interface.

QUESTION ALLOWED, COMMAND ALLOWED:
"Question allowed" indicates if it is legal to request the content of a motor controller register. "Command allowed" indicates if it is legal to allocate a value to the motor controller register. See section *4.2, "Command Description"*, page 29.

FORMAT:
Indicates the default format for the motor controller register.

LONGINT = Signed integer, 32 bit.
WORD = Unsigned integer, 16 bit.
INTEGER = Signed integer, 16 bit.
FIXED = Fixed point * 1000. See section *4.3.9.*
NONE = No data in connection with this command.

Note: By rewriting the table, (carried out by JVL according to customer specifications), the FBG10 can be adapted to other products with another set of commands.

Table.

Question allowed	**Command allowed**	**Format (default)**	**Command name**	**Index**
YES	YES	LONGINT	R00..R99	0..99
YES	YES	WORD	RB00..RB99	100..199
YES	YES	INTEGER	RI00..RI99	200..299
YES	YES	LONGINT	XA00..XA63	300..363
YES	YES	LONGINT	XP00..XP63	364..427
YES	YES	LONGINT	XR00..XR63	428..491
YES	YES	WORD	XV00..XV63	492..555
YES	YES	WORD	CB00..CB63	556..619
YES	YES	LONGINT	RX00..RX31	620..650
YES	YES	LONGINT	AC	651
YES	YES	WORD	ACH	652
YES	YES	LONGINT	ACP	653

5.1 Conversion Table

Question allowed	**Command Allowed**	**Format (default)**	**Command name**	**Index**
YES	YES	WORD	ACT	654
YES	YES	WORD	ADDR	655
YES	NO	WORD	AI1	656
YES	NO	WORD	AI2	657
YES	YES	WORD	AIH1	658
YES	YES	WORD	AIH2	659
NO	NO	NONE	AIL1	660
NO	NO	NONE	AIL2	661
NO	NO	NONE	AIO1	662
NO	NO	NONE	AIO2	663
NO	NO	NONE	AIU1	664
NO	NO	NONE	AIU2	665
YES	YES	WORD	AOUT	666
YES	YES	LONGINT	AP	667
YES	YES	LONGINT	APM	668
YES	YES	LONGINT	APP	669
YES	YES	INTEGER	BIAS	670
YES	YES	FIXED-POINT*100	CA	671
YES	NO	WORD	CFE	672
YES	NO	WORD	CFNE	673
YES	YES	WORD	CHS	674
YES	NO	WORD	CL	675
YES	YES	WORD	CN1	676
YES	YES	WORD	CN2	677
YES	YES	LONGINT	CND1	678
YES	YES	LONGINT	CND2	679
NO	NO	NONE	COMP	680
YES	YES	FIXED-POINT*1000	CON	681

5.1 Conversion Table

Question allowed	**Command allowed**	**Format (default)**	**Command name**	**Index**
YES	YES	FIXED-POINT*10	CP	682
YES	YES	WORD	CS	683
YES	YES	WORD	CT	684
YES	YES	WORD	CTM1	685
YES	YES	WORD	CTM2	686
YES	NO	FIXED-POINT*10	CU	687
YES	NO	FIXED-POINT*100	CUB	688
YES	NO	INTEGER	CV	689
YES	NO	LONGINT	CVI1	690
YES	NO	LONGINT	CVI2	691
YES	NO	LONGINT	CVI3	692
YES	NO	LONGINT	CVI4	693
YES	NO	LONGINT	CVI5	694
YES	NO	LONGINT	CVI6	695
YES	NO	LONGINT	CVI7	696
YES	NO	LONGINT	CVI8	697
NO	NO	LONGINT	D	698
YES	YES	WORD	DIF	699
NO	NO	NONE	E	700
YES	YES	WORD	EP	701
YES	NO	LONGINT	ES0	702
YES	NO	LONGINT	ES1	703
YES	NO	LONGINT	ES2	704
YES	NO	LONGINT	ES3	705
YES	YES	WORD	ET	706
YES	YES	WORD	FEM	707
YES	YES	WORD	FNEM	708

5.1 Conversion Table

Question allowed	**Command allowed**	**Format (default)**	**Command name**	**Index**
YES	YES	FIXED-POINT*1000	GEAR	709
NO	NO	NONE	GO	710
NO	NO	NONE	H	711
YES	YES	WORD	HALL	712
YES	YES	WORD	HL	713
YES	NO	WORD	HM	714
YES	YES	WORD	HML	715
YES	YES	WORD	IL	716
YES	YES	WORD	IMCL	717
YES	NO	BINARY	IN	718
YES	YES	BINARY	INAL	719
YES	YES	WORD	INDEX	720
YES	YES	WORD	JERK	721
YES	YES	WORD	KD	722
YES	YES	WORD	KI	723
YES	YES	WORD	KP	724
YES	YES	FIXED-POINT*1000	KPHASE	725
YES	YES	WORD	KVFF	726
YES	YES	WORD	LED	727
YES	NO	WORD	LINE	728
YES	YES	FIXED-POINT*100	LOAD	729
YES	NO	LONGINT	MCHS	730
YES	YES	WORD	MO	731
YES	YES	WORD	MP	732
NO	NO	NONE	MR	733
NO	NO	NONE	MR0	734
NO	NO	NONE	MR1	735

5.1 Conversion Table

Question allowed	**Command** allowed	**Format** (default)	**Command** name	**Index**
NO	NO	NONE	MR2	736
NO	NO	NONE	MR3	737
NO	NO	NONE	MS	738
NO	NO	NONE	MS0	739
NO	NO	NONE	MS1	740
NO	NO	NONE	MS2	741
YES	YES	WORD	MU	742
YES	NO	WORD	NL	743
YES	YES	WORD	NLL	744
YES	YES	WORD	NLS	745
NO	NO	WORD	NSTART	746
NO	NO	WORD	NSTOP	747
YES	YES	BINARY	OUT	748
YES	YES	LONGINT	PE	749
YES	YES	WORD	PES	750
YES	YES	WORD	PIF	751
YES	NO	WORD	PL	752
YES	YES	WORD	PLL	753
YES	YES	WORD	PLS	754
YES	NO	WORD	PM	755
YES	YES	WORD	PN	756
YES	YES	WORD	PO	757
YES	YES	WORD	POF	758
YES	YES	WORD	POFFSET	759
YES	YES	WORD	POL	760
YES	YES	WORD	PR	761
YES	YES	WORD	PRM	762

5.1 Conversion Table

Question allowed	**Command** allowed	**Format** (default)	**Command** name	**Index**
YES	YES	WORD	PT	763
YES	YES	WORD	PW	764
NO	NO	NONE	RESET	765
YES	NO	WORD	RS	766
NO	NO	NONE	SD	767
NO	NO	NONE	SH	768
YES	NO	WORD	SN	769
YES	YES	WORD	SON	770
YES	YES	LONGINT	SP	771
YES	YES	LONGINT	SPT	772
YES	YES	LONGINT	SR	773
YES	YES	LONGINT	SR2	774
YES	YES	LONGINT	SRA	775
NO	NO	NONE	SR+	776
NO	NO	NONE	SR-	777
NO	NO	NONE	STOP	778
NO	NO	NONE	SZ	779
NO	NO	NONE	SZ+	780
NO	NO	NONE	SZ-	781
YES	YES	WORD	SZT	782
YES	NO	WORD	TQ	783
YES	YES	WORD	TQOUT	784
NO	NO	NONE	UH	785
YES	YES	WORD	VFACTOR	786
YES	YES	WORD	VM	787
YES	NO	WORD	VOL	788
YES	YES	WORD	VS	789

5.1 Conversion Table

Question allowed	Command allowed	Format (default)	Command name	Index
YES	NO	WORD	VV	790
YES	YES	WORD	VVH	791
NO	NO	NONE	VVL	792
NO	NO	NONE	VVO	793
NO	NO	NONE	VVU	794
YES	YES	WORD	ZA	795
YES	YES	INTEGER	ZD	796
YES	YES	WORD	ZL	797
YES	YES	WORD	ZM	798
YES	YES	WORD	ZR	799
YES	NO	WORD	ZS	800
YES	YES	WORD	ZV	801
YES	NO	NONE	Use no parity	4000
YES	NO	NONE	Use odd parity	4001
YES	NO	NONE	Use even parity	4002

5.2 Parity

The FBG10 transmits characters on the RS232 interface with no / odd / even parity. By default, no parity is used.

For example, odd parity can be chosen by sending a question with index 4001 on PROFIBUS. The following commands and questions will then be sent using odd parity. See section *5.1, "Conversion Table"*, page 36.

When receiving characters on the RS232, the FBG10 always ignores a parity-bit.

6 GSD-file

6.1 GSD-file

```
; COM PROFIBUS V 3.3, GSD'-Xport
; Time Stamp: 01/31/00, 12:36:39
#Profibus_DP
; <Unit-Definition-List>
GSD_Revision=1
Vendor_Name="JVL INDSTR"
Model_Name="FBG10"
Revision="0.0"
Ident_Number=0x050C
Protocol_Ident=0
Station_Type=0
Hardware_Release="1.2"
Software_Release="1.2"
9.6_supp=1
19.2_supp=1
93.75_supp=1
187.5_supp=1
500_supp=1
1.5M_supp=1
3M_supp=0
6M_supp=0
12M_supp=0
MaxTsdr_9.6=60
MaxTsdr_19.2=60
MaxTsdr_93.75=60
MaxTsdr_187.5=60
MaxTsdr_500=100
MaxTsdr_1.5M=150
MaxTsdr_3M=250
MaxTsdr_6M=450
MaxTsdr_12M=800
Implementation_Type="SPC3"
Bitmap_Device="DPLINK_"
; Slave-Specification:
Freeze_Mode_supp=0
Sync_Mode_supp=0
Auto_Baud_supp=1
Min_Slave_Intervall=1
Max_Diag_Data_Len=8
Modul_Offset=0
Slave_Family=0
; UserPrmData: Length and Preset:
User_Prm_Data_Len=0
;User_Prm_Data=
; <Module-Definition-List>
Module="Cfg-Data" 0x17,0x27
   EndModule
```

7 Technical data

7.0.1 Specifications:

Supply Voltage	10-30VDC
Current Consumption:	100mA at 24 VDC
RS232 speed:	9600-19200 bps
Fieldbus transfer rate	1.5Mbits
Response time	1ms + RS232 transmission time
Dimensions: HxBxD	74x72x27 mm
Weight:	143 gr.

Order numbers:

FBG10-PROFIBUS-DP
FBG10-INTERBUS
FBG10-CAN
FBG10-DEVICENET

7.0.2 Accessories:

Cable for RS232: 3m	RS232-9-1
Cable for RS232: 0.5m	RS232-9-1-0.5
DSUB plug 9-pole male:	CD0002
DSUB plug 9-pole female:	CD0003
DSUB housing 9-pole:	CD1001
Diskette with Siemens S7 example and gsd file	QE0001
User manual	LB0046-01GB

8 Index