

---

## Profibus DP V1

---

|   |    |
|---|----|
| ■ About this Document .....   | 2  |
| ■ Profibus DPV1 introduction .....  | 3  |
| ■ Principle of data exchange by Profibus DPV1 .....                       | 4  |
| ■ Addressing scheme for DP V1.....  | 5  |
| ■ Accessing VLT parameters on DP V1.....                                  | 5  |
| ■ Description of DP V1 Read / Write Telegram.....                         | 7  |
| ■ Function of Profidrive parameter channel. ....                          | 10 |
| ■ DP V1 on PLC based master.<br>(Master class 1) .....                    | 16 |
| ■ DP V1 on Configuration / commissioning tools.<br>(Master class 2) ..... | 16 |
| ■ Examples Profidrive Parameter access Services. .                        | 30 |
| ■ Example of DP V1 Read Parameter Service. ....                           | 35 |
| ■ Data types. ....  | 37 |
| ■ Size Attributes. ....   | 38 |

---

### ■ About this Document

This document describes the features supported of the Profibus DP V1 functions in all Danfoss Drives with a Profibus option. The functionality is described at a level, which will be sufficient for most system configuration and managing activities.

Beside the drive behavior, a brief description of master applications of master class 1 and master class 2 can be found.

For full system overview, please refer to the following documents:

- Operating Manual for FCM300 (ordering no. MG03BXY), FCD300 (ordering no. MG04BXY), VLT2800 (ordering no. MG28AXY)
- Profibus Manual for FCM300, FCD300, VLT2800 (ordering no. MG90AXY)

where X is referring to the version number code and YY the language code.

For non-Danfoss equipment, please refer to the appropriate specifications.

For further detailed description of this matter, the following documents might be useful:

- Technical Guide “PROFIBUS – DP Extensions to EN 50170 (DPV1)” V2.0, April 1998, Order no. 2.082
- Draft PROFIBUS Profile PROFdrive Profile Drive Technology V3.0 September 2000, Order no. 3.172

Even if you are an experienced Profibus system builder, we suggest you to read this Guide in its entirety before you start programming, since important information can be found in all chapters.

### ■ Assumptions

This guide assumes, that you are using a Danfoss frequency converter with Profibus DP V1 functionality. It is also assumed, that you as a master are using a PLC and/or PC equipped with a Communication card supporting all the Profibus communication services required by your application.

It is assumed, that the following Specifications / limitations are fully respected:

- Profibus DP specifications.
- Profdrive Profile V 2 specifications.
- Profdrive Profile V3 specifications concerning the DP V1 Profdrive parameter channel.
- Limitations in DP V1 features supported

### ■ What you should already know

The Danfoss Profibus is designed to communicate with any master abiding by the Profibus DP standard.

It is therefore assumed, that you have full knowledge of the PC or PLC you intend to use as a master in your system. Any questions pertaining non-Danfoss hardware or software products is beyond the scope of this manual, and is no concern to Danfoss.

If you have questions about how to set up Master – Master connection or communication to non-Danfoss slaves, the appropriate manuals should be consulted.

### ■ Software versions

The following table shows from which software versions Profibus DPV1 is supported. The software version can be read-out in parameter 624 *Software versions*.

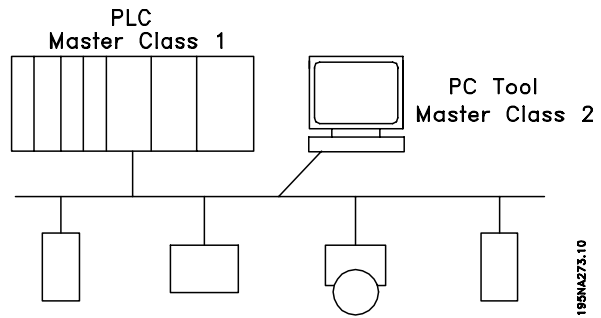
| Unit     | Software version |
|----------|------------------|
| FCD 300  | Ver. 1.30 / 2.00 |
| VLT 2800 | Ver. 2.62 / 2.00 |
| VLT 5000 | Ver. 3.61 / 4.00 |

### ■ Profibus DPV1 introduction

The Profibus DP extension DPV1 offers additional to the cyclical data communication of DP V0, an a-cyclical communication. This feature is possible by a DP master class1 (e.g. PLC) , as well as a DP master class 2 (e.g. PC Tool).

Cyclical communication means that data transfer takes place all the time with a certain update rate. This is the known DP V0 function normally used for quick update of I/O Process Data.

A-cyclical communication means a one time event, mainly used for Read / Write on parameters from Process controllers, PC based tools or monitoring systems.



#### Features of a Master class 1 system:

- Cyclical data exchange (DP V0).
- A-cyclical read/write on parameters.

The a-cyclical connection is fixed , and can not be changed during operation.

In general a master class 1 is used as Process controller, responsible for commands, speed reference, status of the application etc ( PLC or PC based controller.)

The MC 1 a-cyclical connection might be used for general parameter access in the slaves.

#### Features of a Master class 2 system:

- Initiate / Abort a-cyclical connection.
- A-cyclical read/write on parameters.

The a-cyclical connection can dynamically be established (Initiate) or removed (Abort) even though a master class 1 is active on the network.

The MC 2 a-cyclical connection is typically used for configuration or commissioning tools for easy access to each parameter in any slave in the system.

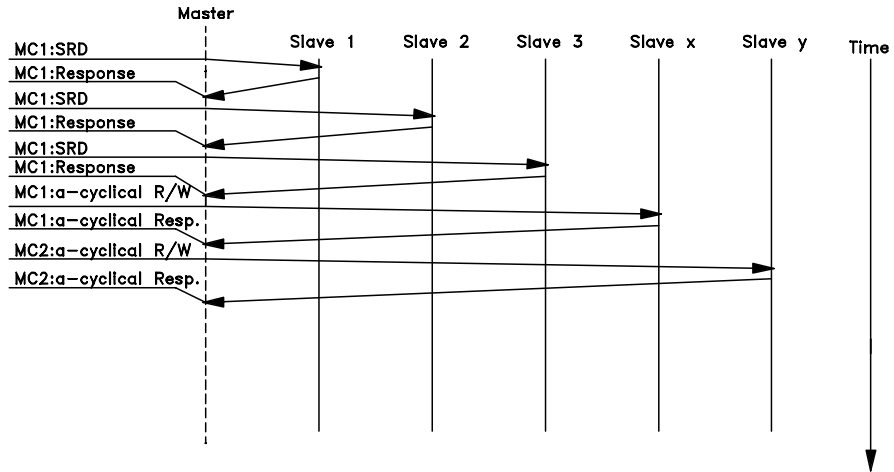
**The frequency converter supports a subset of the V1 specifications shown in the table below.**

| Type of Danfoss Frequency converter | Master type | Read<br>(read data from slave) | Write<br>(read data to slave) | Data transport<br>(read and write data) | Initiate<br>(open a connection) | Abort<br>(close a connection) | Alarm |
|-------------------------------------|-------------|--------------------------------|-------------------------------|---|---------------------------------|-------------------------------|-------|
| FCD 300, VLT2800                    | MC 1        | OK                             | OK                            | -                                       | -                               | -                             | -     |
| FCM 300                             | MC 2        | OK                             | OK                            | -                                       | OK                              | OK                            | -     |
| VLT5000, VLT6000                    | MC 1        | OK                             | OK                            | -                                       | -                               | -                             | -     |
| VLT8000 AQUA                        | MC 2        | OK                             | OK                            | OK                                      | OK                              | OK                            | -     |

### ■ Principle of data exchange by Profibus DPV1

In a DP cycle the MC 1 will first update the cyclical process data for all slaves in the system. After that the MC 1 has the possibility of sending one a-cyclical message to one slave. If a MC 2 is connected, the MC 1 will handle over the Token to

MC 2 who now is aloud to send one a-cyclical message to one slave. After that, the token is handled back to the MC 1, and a new DP cycle is started



MC : Master Class

C1...Cn: Cyclical data

AC1: Acyclical data Master Class 1

AC2: Acyclical data Master Class 2

### ■ Profibus telegram

A DP V1 Read/Write service takes place via a Profibus SD2 telegram, which is a telegram of variable length as shown on the figure below.

DP telegram:

| SD  | LE | Ler | SD  | DA | SA | FC | DSAP | SSAP | DU   | FC S | ED  |
|-----|----|-----|-----|----|----|----|------|------|------|------|-----|
| 68H | x  | X   | 68H | xx | xx | x  | xx   | xx   | x... | xx   | 16H |

|      |                                  |
|------|----------------------------------|
| SD   | Start Delimiter                  |
| LE   | Length                           |
| Ler  | Length repeat                    |
| DA   | Destination Address              |
| SA   | Source Address                   |
| FC   | Function Code                    |
| DSAP | Destination Service Access Point |
| SSAP | Source Service Access Point      |
| DU   | Data Unit for DP services        |
| FCS  | Frame Checking Sequence          |
| ED   | End Delimiter                    |

Profibus DP services is activated via specific Service Access Point's (SAP). For a-cyclical communication, the following SAP are specified:

| Master SAP | Slave SAP         | Meaning   |
|------------|-------------------|---|
| 50 (32H)   | 49 (31H)          | Master Class 2: Initiate request                  |
| 50 (32H)   | 0..48 (0..30H)    | Master Class 2: Abort, Read, Write, Data transfer |
| 51 (33H)   | 50, 51 (32H, 33H) | Master Class 1: Alarm                             |
| 51 (33H)   | 51                | Master Class 1: Read, Write                       |

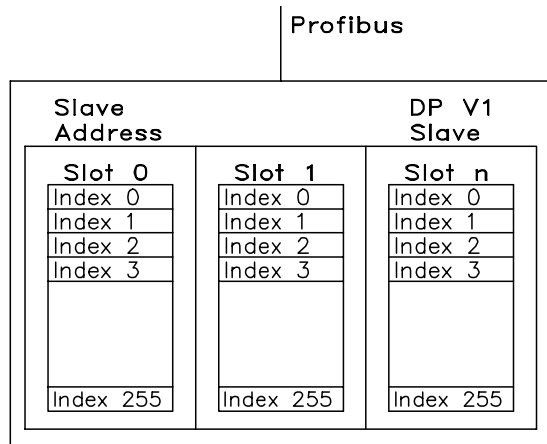
### ■ Addressing scheme for DP V1

The structure of a DP V1 slave is shown in the figure.

A DP V1 slave consists of a number of physical or virtual slots. Slot 0 is always present, and represents the basic unit. Each slot can contain until 255 data blocks addressed by an index and of until 255 byte length.

The master must address a variable in the DP V1 slave as follows:

/Slave address/Module #/Index #



195NA277.10

### ■ Accessing VLT parameters on DP V1

This section describes how DP V1 can be used for accessing VLT parameters.

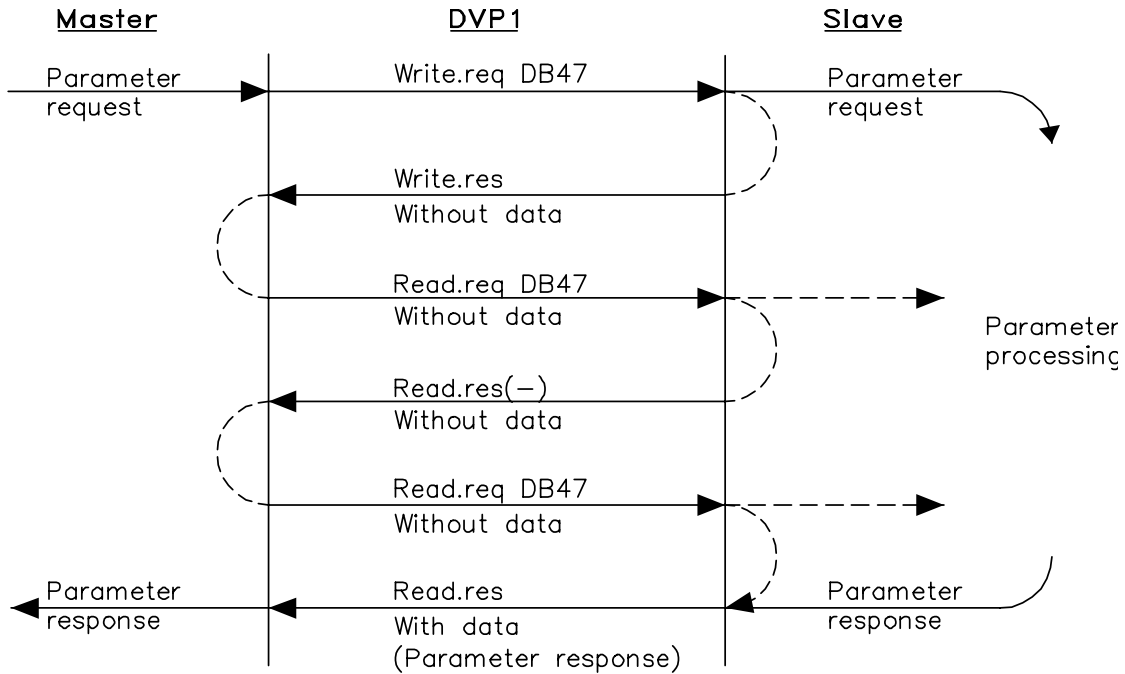
A Danfoss DP V1 slave only contains Slot 0. Because of the complex parameter structure of drives, a specific procedure for accessing parameters for drives is defined. This is named the "Profidrive Parameter Channel", and is part of the Profidrive Profile V3 specifications.

According to this, parameter access must be made through one single DP V1 object with the following address:

Slot = 0.  
Index = 47.

**■ DP V1 Read/write request sequence**

A Read or Write service on a VLT parameter will take place as described in the table below.



195NA278.10

A Read or Write on a VLT parameter must be initiated by a DP V1 write service on index 47 (DB 47). If this Write request is valid, a positive write response without data is returned from the VLT immediately. If not, a negative write response is returned from the VLT. The table on page 12 shows the possible DP V1 response status codes.

The VLT will now interpret the “Profidrive parameter channel” part of the Data Unit, and start to perform this command internal in the VLT.

As the next step, the master will send a Read request. If the VLT is still busy of performing the internal parameter request, a negative response without data is returned from the VLT.

This request will be repeated by the master, until the VLT has the response data ready for the VLT parameter request.

The example shown in section “Example of DP V1 Read Parameter Service”, shows the details of the telegrams needed for the DP V1 Read / Write service.

**■ Description of DP V1 Read / Write Telegram**

The Read / Write telegram has the following principle structure.

| Profibus Telegram Header | Data Unit              |      |      |      |                                 |  |      | Profibus Telegram trailer |  |
|--------------------------|------------------------|------|------|------|---------------------------------|--|------|---------------------------|--|
|                          | DP V1 Command/response |      |      |      | Profidrive V3 Parameter Channel |  |      |                           |  |
|                          | DU 0                   | DU 1 | DU 2 | DU 3 | Req. / Res. Header              |  | Data |                           |  |

The DP V1 command/response part is used for the standard DP V1 read/write on the Slot 0, Index 47 data block.

The Profidrive V 3 Parameter Channel is used for accessing the particular parameter data in the VLT.

In the following the DP V1 command/response handling is described.

**■ Attributes of the DP V1 command / Response part (DU0 through DU3)**

The Names and functions of the attributes of the DP V1 command/response field (DU0 through DU3) is described in the table below.

| DU Byte #           | Value       | Meaning                              | Specified              |
|---------------------|-------------|--------------------------------------|------------------------|
| 0 : Function number | 0x48        | Idle REQ, RES                        | Profibus DP V1         |
|                     | 0x51        | Data transport REQ, RES              |                        |
|                     | 0x56        | Resource Manager REQ                 |                        |
|                     | 0x57        | Initiate REQ, RES                    |                        |
|                     | 0x58        | Abort REQ                            |                        |
|                     | 0x5C        | Alarm REQ, RES                       |                        |
|                     | 0x5E        | Read REQ, RES                        |                        |
|                     | 0x5F        | Write REQ, RES                       |                        |
|                     | 0xD1        | Data transport negative response     |                        |
|                     | 0xD7        | Initiate negative response           |                        |
|                     | 0xDC        | Alarm negative response              |                        |
|                     | 0xDE        | Read negative response               |                        |
|                     | 0xDF        | Write negative response              |                        |
| 1 : Slot #          | Always zero | Slot number in slave                 |                        |
| 2 : Index #         | 47          | Index number in slot / slave         |                        |
| 3 : Data length     | Xx          | Data length of Profidrive data field |                        |
| 4..n                |             | User data                            | PNO Drive Profild V3.0 |

**■ Write Service**

For a DP V1 Write service, the DP V1 Command/Response header must be filled out as shown in the table below.

The DU3 must contain the length of the data field of the Profidrive Parameter Channel

| SD  |       | DU 0           | DU 1          | DU 2        | DU 3                    | DU 4..243          | FCS | ED  |
|-----|-------|----------------|---------------|-------------|-------------------------|--------------------|-----|-----|
| 68H | ..... | fct. no<br>5FH | Slot no.<br>0 | Index<br>47 | Length of<br>data field | Profidrive<br>data | xx  | 16H |

The response of a write service contains no additional data,. Only the DPV1 data DU 0..3 are set. The function number is equal to the request function number, if the response is positive. In case of an negative response, an error code is written in DU2.

Write positive response:

| SD  |       | DU 0           | DU 1          | DU 2        | DU 3        | FCS | ED  |
|-----|-------|----------------|---------------|-------------|-------------|-----|-----|
| 68H | ..... | fct. no<br>5FH | Slot no.<br>0 | Index<br>47 | Length<br>0 | xx  | 16H |

Write negative response:

| SD  |       | DU 0           | DU 1                 | DU 2                              | DU 3 | FCS | ED  |
|-----|-------|----------------|----------------------|-----------------------------------|------|-----|-----|
| 68H | ..... | fct. no<br>DFH | Error De-code<br>80H | Error Class/Code<br>See next page | 0    | xx  | 16H |

**Read Service**

For a DP V1 Read service, the DP V1 Command/Response header must be filled out as shown in the table below.

| SD  |       | DU 0           | DU 1          | DU 2        | DU 3 | FCS | ED  |
|-----|-------|----------------|---------------|-------------|------|-----|-----|
| 68H | ..... | fct. no<br>5EH | Slot no.<br>0 | Index<br>47 | 0    | xx  | 16H |

Only the positive response of a read service contains additional data. In case of an negative response an error code is written in DU2.

Read positive response:

| SD  |       | DU 0           | DU 1          | DU 2        | DU 3                 | DU 4..243          | FCS | ED  |
|-----|-------|----------------|---------------|-------------|----------------------|--------------------|-----|-----|
| 68H | ..... | fct. no<br>5EH | Slot no.<br>0 | Index<br>47 | Length<br>data field | Profidrive<br>data | xx  | 16H |

Read negative response

| SD  |       | DU 0           | DU 1                 | DU 2                              | DU 3 | FCS | ED  |
|-----|-------|----------------|----------------------|-----------------------------------|------|-----|-----|
| 68H | ..... | fct. no<br>DEH | Error De-code<br>80H | Error Class/Code<br>See next page | 0    | xx  | 16H |



**■ DP V1 Class/Error codes**

The table below shows the possible error Classes / codes For the DP V1 Read/write services.

The DU2 will contain the combination of Error Class and Error Code. As an example a “access error / invalid parameter” will be shown as : B8H

| Error Class | Meaning     | Error Code  |
|-------------|-------------|---|
| 10 (0x0A)   | application | 0: read error<br>1: write error<br>2: module failure<br>3 to 7 reserved<br>8: version conflict<br>9:feature not supported<br>10 to 15 user specific   |
| 11 (0x0B)   | access      | 0: invalid index<br>1: write length error<br>2: invalid slot<br>3: type conflict<br>4: invalid area<br>5: state conflict<br>6: access denied<br>7: invalid range<br>8: invalid parameter<br>9: invalid type<br>10 to 15 user specific |
| 12 (0x0C)   | resource    | 0: read constrain conflict<br>1: write constrain conflict<br>2: resource busy<br>3: resource unavailable<br>4 to 7 reserved<br>10 to 15 user specific   |

**■ Function of Profidrive parameter channel**

The table below shows the structure of the Profidrive Parameter channel. By this it is possible to access the following VLT parameter values and attributes:

- Parameter values of simple variable, array and visible string.

- Parameter description elements such as type, Min./max. value etc.
- Describing text to parameter values.

Beside this it is possible to access multiple parameters in one telegram.

Profibus DP V1 telegram for Read/write on VLT parameter.

|                          |                        |      |      |      |                                 |      |  |                           |  |
|--------------------------|------------------------|------|------|------|---------------------------------|------|--|---------------------------|--|
| Profibus Telegram Header | Data Unit              |      |      |      |                                 |      |  | Profibus Telegram trailer |  |
|                          | DP V1 Command/response |      |      |      | Profidrive V3 Parameter Channel |      |  |                           |  |
|                          | DU 0                   | DU 1 | DU 2 | DU 3 | Req. / Res. Header              | Data |  |                           |  |

The Profidrive Parameter Channel part of the Data Unit is responsible of accessing the particular parameter in the VLT

The table below shows the principle structure of the Profidrive Parameter Channel.

The DP V1 Parameter Request telegram consists of 3 blocks of data:

- A Request Header, which defines the kind of request (Read or Write), and the number of parameters to access. The Request Reference is set by the master, which uses this information to evaluate the response.
- An address field, where all addressing attributes of the desired parameters are defined.
- A Data field, where all parameter data values are placed

DP V1 Parameter Request

Byte

|   |                   |                  |       |
|---|-------------------|------------------|-------|
| Request Header  | Request Reference | ID               | 0 / 1 |
|   | Axis              | No. of Parameter | 2 / 3 |
|   | Attribute         | No. of Elements  | 4 / 5 |
|   | Parameter Number  |                  | 6 / 7 |
|   | Subindex          |                  | 8 / 9 |
| -   |                   |                  |       |
| -   |                   |                  |       |
| nth Parameter Address   |                   |                  |       |
| ...   |                   |                  |       |
| 1st Parameter Value(s)<br>(only for change parameter request) | Format            | No. of Values    |       |
|   | Values            |                  |       |
| -   |                   |                  |       |
| nth Parameter Values  |                   |                  |       |
| ...   |                   |                  |       |

The DP V1 Parameter response telegram consists of 2 blocks of data:

- A response header which indicates if the request is performed without errors (Response ID), the number of parameters, and the Request Reference set by the master within the corresponding request telegram.
- A Data field, where the requested data are placed. If one or more internal requests has failed, an Error Code is placed instead of the data values.

DPV1 parameter response:

|   |                            |                   | Byte  |
|---|----------------------------|-------------------|-------|
| Response Header   | Request Reference mirrored | Response ID       | 0 / 1 |
|   | Axis mirrored              | No. of Parameters | 2 / 3 |
| 1st Parameter Value(s)<br>(only for request<br>"Request") | Format                     | No. of Values     | 4 / 5 |
|   | Values or Error Values     |                   | 6     |
|   | ...                        |                   |       |
| -   |                            |                   |       |
| nth Parameter Values ...                                  | ...                        |                   |       |

As the response telegram do not include parameter addressing information, the master must identify the structure of the response data from the request telegram.

On page 30 various examples for Parameter access are shown.

**■ Request / Response Attributes**

The table contains an overview of the possible attributes of the Profidrive parameter channel.

| Field               | Datatype   | Values   | Remark  |
|---------------------|------------|--|---|
| Request reference   | Unsigned8  | 0x01..0xFF   |   |
| Request ID          | Unsigned8  | 0x01 request par value<br>0x02 change par value  | Identifier for read or write request                  |
| Response ID         | Unsigned8  | 0x01 request parameter(+)<br>0x02 change parameter(+)<br>0x81 request parameter(-)<br>0x82 change parameter(-) | Identifier for the response                           |
| Axis                | Unsigned8  | 0x00..0xFF number (always 0)   |   |
| Number of Parameter | Unsigned8  | 0x01..0x25 Quantity 1..37  | Limitation : DPV1-telegram length                     |
| Attribute           | Unsigned8  | 0x10 value<br>0x20 description<br>0x30 text  | Describing the kind of data.                          |
| Number of elements  | Unsigned8  | 0x00 special function<br>0x01..0x75 Quantity 1..117  | Limitation : DPV1-telegram length                     |
| Parameter number    | Unsigned16 | 0x0001... number 1..65535<br>0xFFFF  | VLT parameter number.                                 |
| Subindex            | Unsigned16 | 0x0000... number 0..65535<br>0xFFFF  | For pointing out array ,text and description elements |
| Format              | Unsigned8  | 0x01..0x36 datatypes<br>0x40 Null<br>0x41 Byte<br>0x42 Word<br>0x43 doubleword<br>0x44 error                   |   |
| Number of values    | Unsigned8  | 0x00..0xEA Quantity 0..234   | Limitation : DPV1-telegram length                     |
| Error Number        | Unsigned16 | 0x0000... Error number, see<br>0x00FF appendix   |   |

In the following the attributes for the Request/ response headers are described.

**Request Reference**

Unique identification of request/response pair for the master. The master changes the request reference with each new request. The slave mirrors the request reference in the response.

**Request ID**

Two request identifications are defined:

- Request VLT parameter = 01H (read)
- Change VLT parameter = 02H (write)

A parameter change is in general stored in volatile memory. (RAM). Via parameter 971 the data can be stored in non volatile memory (EEPROM).

**Response ID**

Mirroring of the request ID with supplement information whether the request was executed positively(+) or negatively(-). If the response is negative, error codes are entered per partial response instead of values.

**Axis**

Always zero.

**No. of Parameters**

For multi parameter request specifying the number of the Parameter Address and/or Parameter Value areas. For single request the number is 1. The value range for multiple request is 1..37.

**■ Attribute**

The attribute determines which kind of data to access.

The VLT will respond on the attributes Value (10H) , Description (20H) and Text (30H)

Attribute Value (10H)

The attribute value allows to read or write parameter values.

Attribute Description (20H)

The attribute description allows to read parameter description.

It is possible to read out one single description element, or all elements for one parameter in one telegram.

The table below shows a overview of the existing Parameter Description which exists for each parameter in the VLT.

Parameter description elements (all elements are read-only) :

| Subindex | Meaning                                      | Data Type        |
|----------|--|------------------|
| 1        | Identifier ID                                | V2               |
| 2        | Number of array elements or length of string | U16              |
| 3        | Standardization factor                       | float            |
| 4        | Variable attribute                           | Octetstring 2    |
| 5        | reserved                                     | Octetstring 4    |
| 6        | Name   | visiblestring 16 |
| 7        | Lower limit                                  | Integer 32 *     |
| 8        | Upper limit                                  | Integer 32 *     |
| 9        | reserved                                     | Octetstring 2    |
| 10       | ID extension                                 | V2               |
| 11       | PZD reference parameter (not supported)      | U16              |
| 12       | PZD normalization (not supported)            | V2               |
| 0        | Complete description                         | Octetstring 46   |

\* The actual format of Upper/Lower Limits is equal to the format of the actual parameter value, and can be determined by read of the Identifier ID and the Variable attribute of the actual parameter.

In the following each description element is described.

**Identifier ID**

Additional characteristic of a parameter.

| Bit  | Meaning  |
|------|--|
| 15   | reserved   |
| 14   | array  |
| 13   | par value can be reset only                                |
| 12   | par has been changed from factory setting                  |
| 11   | reserved   |
| 10   | additional text array available                            |
| 9    | parameter is read only                                     |
| 8    | standardization factor and variable attribute not relevant |
| 7..0 | data type (see appendix)                                   |

**Number of array elements**

contains the amount of array elements, if the parameter is an array, the string length, if parameter value is a string or 0 if the parameter is none of both.

**Standardization factor**

This function is not supported. The value 0 is returned (Float format)

**Variable attribute**

Consists of 2 bytes, the first byte contains the variable index, the second the conversion index, refer to the section "Size Attributes".

**Name**

contains the parameter name, limited to 16 characters, e.g. "LANGUAGE" for parameter 1. This text is available in the language selected in par 1.

**Lower limit**

contains the min value of the parameter. Format is equal to the format of the actual parameter.

**Upper limit**

contains the max value of the parameter. Format is equal to the format of the actual parameter.

**ID extension**

Not supported

**PZD Reference parameter**

not supported

**Field PZD normalization**

not supported

**Complete description**

this delivers back the complete parameter description with the fields 1..12 in order. Length = 46 byte.

For not supported elements, the value zero will be returned.

Attribute Text (30H)

For some VLT parameters a describing text to parameter value exists, which can be read out by this attribute. The availability of a text description for a parameter is indicated by a bit set in the Identifier (ID) Parameter Description element, which can be read out by the Description Attribute (20H) subindex = 1. If bit 10 is set, a describing text exists for each value of the parameter.

As an example parameter 01 (Language) has the setting values 0 trough 5. For each of these values a specific text exists: (0 = ENGLISH, 2 = DEUTSCH etc.).

**Format**

Format specify the type of each parameter (word, byte, etc.), see below

**Values**

If the values consist of an odd number of bytes, a zero byte is appended in order to secure the word structure of the telegrams.

In the case of a positive partial response, the parameter value contains the following:

- Format = (Data Type or Byte, Word, Double Word)
- Number of values
- the values

In the case of a negative partial response, the parameter value contains the following:

- Format = error (44H)
- No. of values = 1
- Value = error value = error number

Error Number for Drive Profile V3.0

By a invalid parameter request, the VLT will return the corresponding error code. The table below shows the possible error codes.

error numbers for DPV1 parameter requests

| Error number | Meaning  | Additional Info |
|--------------|--|-----------------|
| 0x00         | unknown parameter  | 0               |
| 0x01         | parameter is read-only   | subindex        |
| 0x02         | value out of range due to max/min value  | subindex        |
| 0x03         | wrong subindex   | subindex        |
| 0x04         | par is no array  | 0               |
| 0x05         | wrong datatype (wrong data length)   | 0               |
| 0x06         | it is not allowed to set this par (only reset)   | subindex        |
| 0x07         | descriptive element is read-only   | subindex        |
| 0x09         | no description available (only value)  | 0               |
| 0x0b         | process control not possible   | 0               |
| 0x0f         | no text array available (only value)   | 0               |
| 0x11         | not possible in current state  | 0               |
| 0x14         | value out of range due to drive state/configuration  | subindex        |
| 0x15         | reply too long (more than 240 bytes)   | 0               |
| 0x16         | wrong parameter address (unknown or unsupported value for attribute, element, par number or subindex or illegal combination) | 0               |
| 0x17         | illegal format (for writing)   | 0               |
| 0x18         | value amount not consistent  | 0               |
| 0x65         | wrong axis : action not possible with this axis  | -               |
| 0x66         | unknown service request  | -               |
| 0x67         | this service is not possible with multi parameter access   | -               |
| 0x68         | parameter value can not be read from bus   | -               |

### ■ Profibus DPV1 on a Siemens S7 PLC

This chapter describes how to set up a Profibus DPV1 communication between a Danfoss VLT frequency converter and S7 315-2 DP PLC from Siemens. The Profibus DPV1 telegrams are in these examples handle by the S7 315-2 DP PLC. It is assumed that you are already familiar with the Simatic S7 system.

This chapter describes:

- Hardware configuration
- Function block FB 36 "PDAT\_AC2"
- Single read and write commands
- Multiple parameter download
- Extended diagnose



#### NOTE!:

The examples do not describe all the functions needed for a real application, for example error handling.

Only the steps that are required in order to establish communication between the VLT frequency converter and the S7 PLC are shown.

Details of some of the components/software:

- FCD 300 with DPV1
- PLC S7-315-2 DP version 1.1
- S7 Manager version 5.1
- Siemens Drive ES Simatic

### ■ Hardware configuration

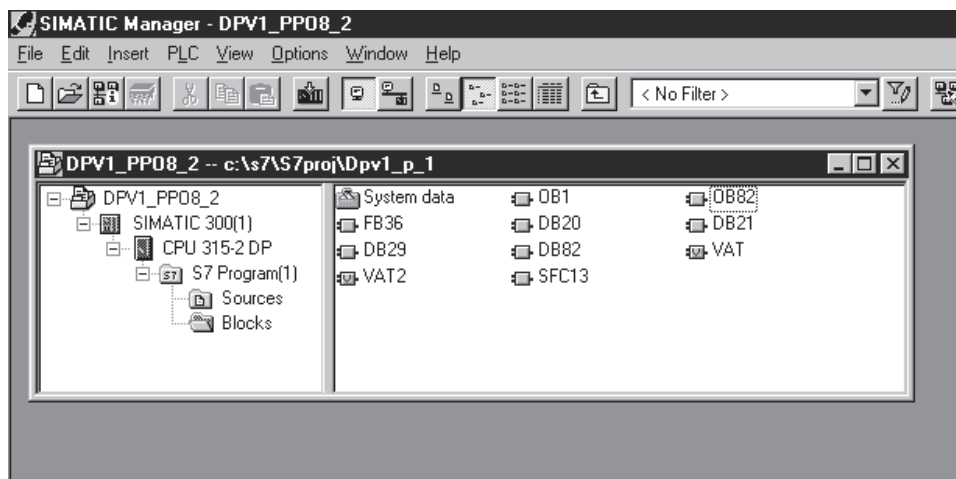
First step is to import a new GSD/GSE of FCD 300 into Siemens S7 Manager.

A GSD/GSE files of our VLT frequency converters must be installed to support DPV1. Please check your GSD/GSE in the list on page 36.

The GSD/GSE file can be ordered from your local Danfoss representative or it can be downloaded at [www.danfossdrives.com](http://www.danfossdrives.com).

Start a new project and click on Simatic 300 station. Now double click on Hardware.

In Hardware the Profibus DP network is configured.



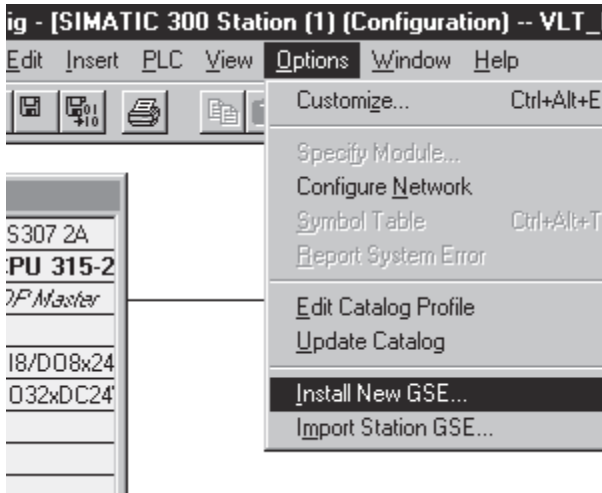
- OB1: Contains a call to FB 36 "PDAT\_AC2"
- OB 82: Is used to the extended diagnose. Contains SFC 13 "DPNRM\_DG"
- FB36: Pre-define Function block from Siemens which handle Profibus DPV1 telegrams.
- DB20: Request (send) telegrams for DPV1.
- DB21: Receive (response) telegrams for DPV1.

- DB29: This data block is used together with FB 36, which handles the communication.
- DB82: This data block is used together with OB 82 Extended diagnose.
- VAT: Variable Table.
- SFC13: A Special Function Call that is used together with extended diagnose.



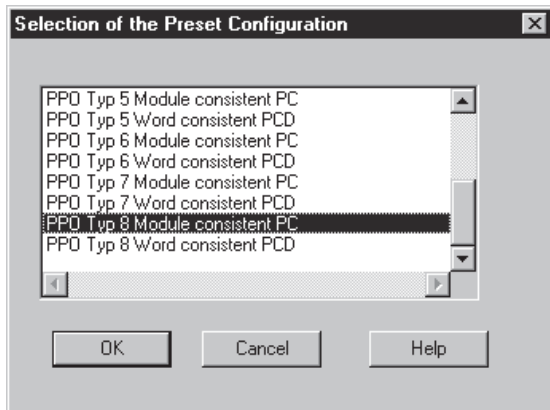
### ■ Step 1 Install GSD/GSE file

Go to *Option* and choose *Install New GSE* to read in the new GSD/GSE to the "Hardware Catalog".



The "Hardware Catalog" must be updated before the new GSD/GSE file is active. Go to *Option* again and choose *Update Catalog*. The new GSD/GSE file is now available in the "Hardware Catalog".

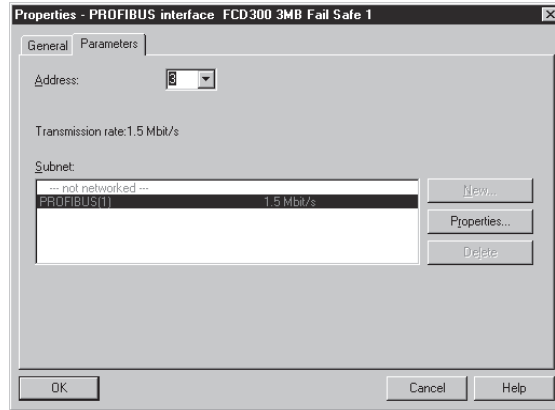
Drag and drop the FCD 300 from the "Hardware Catalog" to the Profibus network. Chose a PPO type and click on OK.



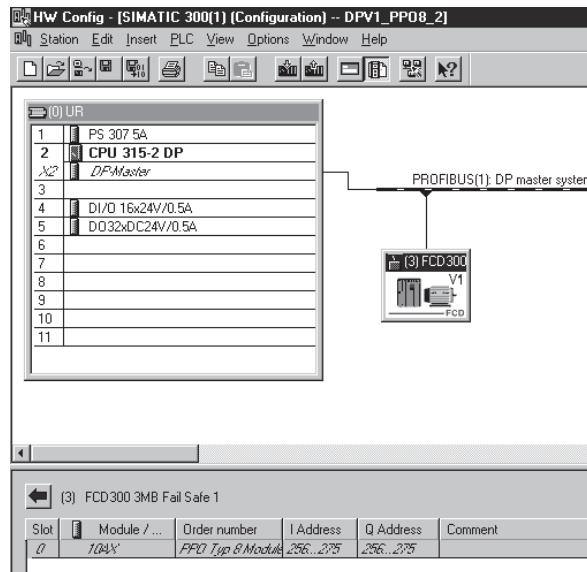
The selected PPO type can be seen in parameter *904 PPO type* after the Master have initialise the communication.

Module consistent data means that all the data exchange between the PLC and the Profibus master is transfer at one time (module consistent). By Word consistent the data exchange between the PLC and Profibus master is consistent in words. The first 4 words of PPO type 1, type 2 and type 5 consist of a PCA part (Parameter Characteristic) and this is always module consistent.

Next is the setting of the VLT frequency converter address. The choice here must correspond to the setting in VLT parameter *918 Station address*. Note that a change in parameter *918 Station address* is first valid at next power up.



The FCD 300 can now be seen on the Profibus network.



Select an I/O address in your Simatic S7 PLC. By double clicking the first module (4AX in the DP-Slave) in the Slave Station, you can either select the address in the Simatic S7 PLC or use the default addresses.

In this example we have chosen the start address to be 256 dec. and a PPO type 8 for communication with FCD 300.

Download the configuration to the PLC.

■ **Function block FB 36 "PDAT\_AC2"**

FB 36 "PDAT\_AC2" is a pre-define function block from Siemens that can handle the DPV1 communication. FB 36 "PDAT\_AC2" uses a pre-define datablock DB29.

In this example FB 36 is placed in OB1.

LADDR is the address in the I/O area where the VLT frequency is mapped. In this example the address starts at 256 dec. (100 Hex).

START will trigger FB 36 and starts a DPV1 telegram.

DB\_NO\_OR is the datablock for sending telegrams and DB\_NO\_AN is the response. In OFFSET\_OR and OFFSET\_AN a offset in the datablocks can be define.

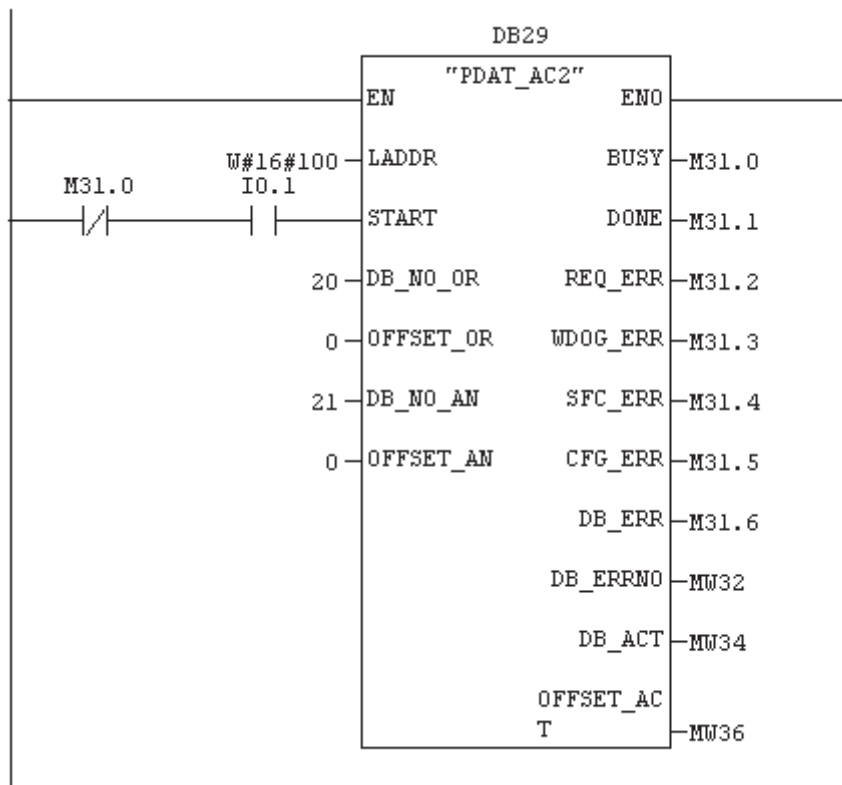
In this example Datablock DB20.0 is the start of sending a telegram to the VLT frequency converter and Datablock DB21.0 is the where the response is. Please refer to S7 Manager to get a description on how a Datablock is made.

The BUSY bit indicates that a DPV1 telegram is active.

The DONE bit shows that the DPV1 communication is finished.

The rest of the bits are error bits if the communication was interrupted.

---



### ■ Single parameter read

This example shows a single parameter read command of parameter 207 *Ramp up time 1*. The ramp up time is 10.00 sec.

Please also see the chapter *Description of DP V1 Read/Write Telegram* to see how the telegram is built up.

|    | Address                 | Disp | Status    | value     | Modify value |
|----|-------------------------|------|-----------|-----------|--------------|
| 1  | // Send data.           |      |           |           |              |
| 2  | // Req ref              |      |           |           |              |
| 3  | DB20.DBB 0              | HEX  | B#16#03   | B#16#03   |              |
| 4  | // Req ID               |      |           |           |              |
| 5  | DB20.DBB 1              | HEX  | B#16#01   | B#16#01   |              |
| 6  | // Axis                 |      |           |           |              |
| 7  | DB20.DBB 2              | HEX  | B#16#01   | B#16#01   |              |
| 8  | // Number of parameters |      |           |           |              |
| 9  | DB20.DBB 3              | HEX  | B#16#01   | B#16#01   |              |
| 10 | // Attribute            |      |           |           |              |
| 11 | DB20.DBB 4              | HEX  | B#16#10   | B#16#10   |              |
| 12 | // Number of elements   |      |           |           |              |
| 13 | DB20.DBB 5              | HEX  | B#16#01   | B#16#01   |              |
| 14 | // Parameter number     |      |           |           |              |
| 15 | DB20.DBW 6              | HEX  | W#16#00CF | W#16#00CF |              |
| 16 | // Subindex             |      |           |           |              |
| 17 | DB20.DBW 8              | HEX  | W#16#0000 | W#16#0000 |              |
| 18 | // Data format          |      |           |           |              |
| 19 | DB20.DBB 10             | HEX  | B#16#07   | B#16#07   |              |
| 20 | // No of values         |      |           |           |              |
| 21 | DB20.DBB 11             | HEX  | B#16#01   | B#16#01   |              |
| 22 | // Values               |      |           |           |              |

Request telegram

The request reference is handle by FB 36.  
01 Hex for a read command

01 Hex for a single parameter

10 Hex for a Value

CF Hex for parameter 207

|    |                   |     |           |  |  |
|----|-------------------|-----|-----------|--|--|
| 28 | // Response       |     |           |  |  |
| 29 | // Req. ref.      |     |           |  |  |
| 30 | DB21.DBB 0        | HEX | B#16#03   |  |  |
| 31 | //Req ID          |     |           |  |  |
| 32 | DB21.DBB 1        | HEX | B#16#01   |  |  |
| 33 | // No of Axis.    |     |           |  |  |
| 34 | DB21.DBB 2        | HEX | B#16#01   |  |  |
| 35 | // Number of Par. |     |           |  |  |
| 36 | DB21.DBB 3        | HEX | B#16#01   |  |  |
| 37 | //Values          |     |           |  |  |
| 38 | DB21.DBW 4        | HEX | W#16#0701 |  |  |
| 39 | DB21.DBW 6        | HEX | W#16#0000 |  |  |
| 40 | DB21.DBW 8        | HEX | W#16#03E8 |  |  |
| 41 | DB21.DBW 10       | HEX | W#16#0000 |  |  |
| 42 | DB21.DBW 12       | HEX | W#16#0000 |  |  |
| 43 | DB21.DBW 14       | HEX | W#16#0000 |  |  |

Response telegram

07 Hex = Datatype unsigned 32  
01 Hex read date OK  
03E8 Hex = 1000 dec corresponding to 10.00 sec.

### ■ Single parameter write (Byte)

This example shows a single parameter write command to parameter 302 *Digital input 18*. The value of parameter 302 *Digital input 18* is set to *Start [7]*.

Please also see the chapter *Description of DP V1 Read/Write Telegram* to see how the telegram is built up.

|    | Address                 | Disp | Status value | Modify value |
|----|-------------------------|------|--------------|--------------|
| 1  | // Send data.           |      |              |              |
| 2  | // Req ref              |      |              |              |
| 3  | DB20.DBB 0              | HEX  | B#16#03      | B#16#03      |
| 4  | // Req ID               |      |              |              |
| 5  | DB20.DBB 1              | HEX  | B#16#02      | B#16#02      |
| 6  | // Axis                 |      |              |              |
| 7  | DB20.DBB 2              | HEX  | B#16#01      | B#16#01      |
| 8  | // Number of parameters |      |              |              |
| 9  | DB20.DBB 3              | HEX  | B#16#01      | B#16#01      |
| 10 | // Attribute            |      |              |              |
| 11 | DB20.DBB 4              | HEX  | B#16#10      | B#16#10      |
| 12 | // Number of elements   |      |              |              |
| 13 | DB20.DBB 5              | HEX  | B#16#01      | B#16#01      |
| 14 | // Parameter number     |      |              |              |
| 15 | DB20.DBW 6              | HEX  | W#16#012E    | W#16#012E    |
| 16 | // Subindex             |      |              |              |
| 17 | DB20.DBW 8              | HEX  | W#16#0000    | W#16#0000    |
| 18 | // Data format          |      |              |              |
| 19 | DB20.DBB 10             | HEX  | B#16#05      | B#16#05      |
| 20 | // No of values         |      |              |              |
| 21 | DB20.DBB 11             | HEX  | B#16#01      | B#16#01      |
| 22 | // Values               |      |              |              |
| 23 | DB20.DBW 12             | HEX  | W#16#0700    | W#16#0700    |
| 24 | DB20.DBW 14             | HEX  | W#16#0000    | W#16#0000    |
| 25 | DB20.DBW 16             | HEX  | W#16#0000    |              |
| 26 | DB20.DBW 18             | HEX  | W#16#0000    |              |
| 27 |                         |      |              |              |

Request telegram

The request reference is handle by FB 36.

02 Hex for a write command

01 Hex for a single parameter

10 Hex for a Value

12E Hex for parameter 302

05 Hex as parameter 302 is a Unsigned 8

07 Hex is start [7] in parameter 302.



**NOTE!:**  
Please note as this is a Unsigned 8 the value is align left.

### ■ Single parameter write (long word)

This example shows a single write command to parameter 207 *Ramp up time 1*. The value of parameter 207 is set to 800.00 sec.

Please also see the chapter *Description of DP V1 Read/Write Telegram* to see how the telegram is built up.

|    | Address                 | Disp | Status value | Modify value |  |
|----|-------------------------|------|--------------|--------------|--|
| 1  | // Send data.           |      |              |              |  |
| 2  | // Req ref              |      |              |              | Request telegram                                   |
| 3  | DB20.DBB 0              | HEX  | B#16#03      | B#16#03      | The request reference is handle by FB 36.          |
| 4  | // Req ID               |      |              |              |  |
| 5  | DB20.DBB 1              | HEX  | B#16#02      | B#16#02      | 02 Hex for a write command                         |
| 6  | // Axis                 |      |              |              |  |
| 7  | DB20.DBB 2              | HEX  | B#16#01      | B#16#01      |  |
| 8  | // Number of parameters |      |              |              |  |
| 9  | DB20.DBB 3              | HEX  | B#16#01      | B#16#01      | 01 Hex for a single parameter                      |
| 10 | // Attribute            |      |              |              |  |
| 11 | DB20.DBB 4              | HEX  | B#16#10      | B#16#10      | 10 Hex for a Value                                 |
| 12 | // Number of elements   |      |              |              |  |
| 13 | DB20.DBB 5              | HEX  | B#16#01      | B#16#01      |  |
| 14 | // Parameter number     |      |              |              |  |
| 15 | DB20.DBW 6              | HEX  | W#16#00CF    | W#16#00CF    | CF Hex for parameter 207                           |
| 16 | // Subindex             |      |              |              |  |
| 17 | DB20.DBW 8              | HEX  | W#16#0000    | W#16#0000    |  |
| 18 | // Data format          |      |              |              |  |
| 19 | DB20.DBB 10             | HEX  | B#16#07      | B#16#07      | 07 Hex as parameter 207 is a Unsigned 32           |
| 20 | // No of values         |      |              |              |  |
| 21 | DB20.DBB 11             | HEX  | B#16#01      | B#16#01      |  |
| 22 | // Values               |      |              |              |  |
| 23 | DB20.DBW 12             | HEX  | W#16#0001    | W#16#0001    | 13880 Hex = 80000 dec corresponding to 800.00 sec. |
| 24 | DB20.DBW 14             | HEX  | W#16#3880    | W#16#3880    |  |
| 25 | DB20.DBW 16             | HEX  | W#16#0000    |              |  |
| 26 | DB20.DBW 18             | HEX  | W#16#0000    |              |  |
| 27 |                         |      |              |              |  |

### ■ Single parameter write - error response

This example shows a single parameter write command to parameter 207 *Ramp up time 1* where the format is wrong. The format of parameter 207 is 7. In this example the format is set to 6 and the error response back 05 Hex means wrong datatype.

Please also see the chapter *Description of DP V1 Read/Write Telegram* to see how the telegram is built up.

|    | Address             | Disp | Status value | Modify value |   |
|----|---------------------|------|--------------|--------------|---|
| 14 | // Parameter number |      |              |              |   |
| 15 | DB20.DBW 6          | HEX  | W#16#00CF    | W#16#00CF    | CF Hex for parameter 302                        |
| 16 | // Subindex         |      |              |              |   |
| 17 | DB20.DBW 8          | HEX  | W#16#0000    | W#16#0000    |   |
| 18 | // Data format      |      |              |              |   |
| 19 | DB20.DBB 10         | HEX  | B#16#06      | B#16#06      | The format is here set to 06 Hex which is wrong |
| 20 | // No of values     |      |              |              |   |
| 21 | DB20.DBB 11         | HEX  | B#16#01      | B#16#01      |   |
| 22 | // Values           |      |              |              |   |
| 23 | DB20.DBW 12         | HEX  | W#16#0000    | W#16#0000    |   |
| 24 | DB20.DBW 14         | HEX  | W#16#03E8    | W#16#03E8    |   |
| 25 | DB20.DBW 16         | HEX  | W#16#0000    |              |   |
| 26 | DB20.DBW 18         | HEX  | W#16#0000    |              |   |
| 27 |                     |      |              |              |   |
| 28 | // Responce         |      |              |              |   |
| 29 | // Req. ref.        |      |              |              |   |
| 30 | DB21.DBB 0          | HEX  | B#16#03      |              |   |
| 31 | //Req ID            |      |              |              |   |
| 32 | DB21.DBB 1          | HEX  | B#16#82      |              | 82 Hex means change value error                 |
| 33 | // No of Axis.      |      |              |              |   |
| 34 | DB21.DBB 2          | HEX  | B#16#01      |              |   |
| 35 | // Number of Par.   |      |              |              |   |
| 36 | DB21.DBB 3          | HEX  | B#16#01      |              |   |
| 37 | //Values            |      |              |              |   |
| 38 | DB21.DBW 4          | HEX  | W#16#4401    |              | 44 Hex means error                              |
| 39 | DB21.DBW 6          | HEX  | W#16#0005    |              | 05 Hex means wrong datatype                     |
| 40 | DB21.DBW 8          | HEX  | W#16#0000    |              |   |
| 41 | DB21.DBW 10         | HEX  | W#16#0000    |              |   |
| 42 | DB21.DBW 12         | HEX  |              |              |   |

DPV1\_PP08\_2\SIMATIC 300(1)\...\S7 Program(1)

### ■ Single parameter write (array element)

This example shows a single parameter write command to parameter 916 *PCD Read configuration* index number 1. The value in index number 1 is set to 520.

Please also see the chapter *Description of DP V1 Read/Write Telegram* to see how the telegram is built up.

| Monitoring and Modifying Variables - [@VAT -- DPV1_PP08 |                         |      |              |              |   |
|---|-------------------------|------|--------------|--------------|---|
| Table Edit Insert PLC Variable View Options Window H    |                         |      |              |              |   |
|   | Address                 | Disp | Status value | Modify value |   |
| 2   | // Req ref              |      |              |              | Request telegram                          |
| 3   | DB20.DBB 0              | HEX  | B#16#03      | B#16#03      | The request reference is handle by FB 36. |
| 4   | // Req ID               |      |              |              | 02 Hex for a write command                |
| 5   | DB20.DBB 1              | HEX  | B#16#02      | B#16#02      |   |
| 6   | // Axis                 |      |              |              |   |
| 7   | DB20.DBB 2              | HEX  | B#16#01      | B#16#01      |   |
| 8   | // Number of parameters |      |              |              | 01 Hex for a single parameter             |
| 9   | DB20.DBB 3              | HEX  | B#16#01      | B#16#01      |   |
| 10  | // Attribute            |      |              |              | 10 Hex for a Value                        |
| 11  | DB20.DBB 4              | HEX  | B#16#10      | B#16#10      |   |
| 12  | // Number of elements   |      |              |              |   |
| 13  | DB20.DBB 5              | HEX  | B#16#01      | B#16#01      |   |
| 14  | // Parameter number     |      |              |              | 0394 Hex for parameter 916                |
| 15  | DB20.DBW 6              | HEX  | W#16#0394    | W#16#0394    |   |
| 16  | // Subindex             |      |              |              | 0000 Hex for index 1, see note            |
| 17  | DB20.DBW 8              | HEX  | W#16#0000    | W#16#0000    |   |
| 18  | // Data format          |      |              |              | 06 Hex as parameter 916 is a Unsigned 16  |
| 19  | DB20.DBB 10             | HEX  | B#16#06      | B#16#06      |   |
| 20  | // No of values         |      |              |              |   |
| 21  | DB20.DBB 11             | HEX  | B#16#01      | B#16#01      |   |
| 22  | // Values               |      |              |              | 208 Hex = 520 dec.                        |
| 23  | DB20.DBW 12             | HEX  | W#16#0208    | W#16#0208    |   |
| 24  | DB20.DBW 14             | HEX  | W#16#0000    | W#16#0000    |   |
| 25  | DB20.DBW 16             | HEX  | W#16#0000    |              |   |
| 26  | DB20.DBW 18             | HEX  | W#16#0000    |              |   |
| 27  |                         |      |              |              |   |



#### NOTE:

Note that sub-indexes starts in the VLT with index 1 and the Profibus DPV1 starts with index 0 i.e. that a VLT parameter sub-index 1 is equal to DPV1 index 0.



### ■ Multiple parameter read

This example shows a multiple read command to parameter 102 *Motor power*, parameter 103 *Motor voltage* and parameter 104 *Motor frequency*.

Please also see the chapter *Description of DP V1 Read/Write Telegram* to see how the telegram is built up.

|    | Address                 | Disp | Status value | Modify value |  |
|----|-------------------------|------|--------------|--------------|--|
| 5  | DB20.DBB 1              | HEX  | B#16#01      | B#16#01      | Request telegram                             |
| 6  | // Axis                 |      |              |              | 01 Hex for a read command                    |
| 7  | DB20.DBB 2              | HEX  | B#16#01      | B#16#01      |  |
| 8  | // Number of parameters |      |              |              |  |
| 9  | DB20.DBB 3              | HEX  | B#16#03      | B#16#03      | 03 Hex for reading three parameters          |
| 10 | // Attribute            |      |              |              |  |
| 11 | DB20.DBB 4              | HEX  | B#16#10      | B#16#10      | 10 Hex for a Value                           |
| 12 | // Number of elements   |      |              |              |  |
| 13 | DB20.DBB 5              | HEX  | B#16#01      | B#16#01      |  |
| 14 | // Parameter number     |      |              |              |  |
| 15 | DB20.DBW 6              | HEX  | W#16#0066    | W#16#0066    | 0066 Hex for parameter 102                   |
| 16 | // Subindex             |      |              |              |  |
| 17 | DB20.DBW 8              | HEX  | W#16#0000    | W#16#0000    |  |
| 18 | // Attribute            |      |              |              |  |
| 19 | DB20.DBB 10             | HEX  | B#16#10      | B#16#10      |  |
| 20 | // Number of elements   |      |              |              |  |
| 21 | DB20.DBB 11             | HEX  | B#16#01      | B#16#01      |  |
| 22 | // Parameter number     |      |              |              |  |
| 23 | DB20.DBW 12             | HEX  | W#16#0067    | W#16#0067    | 0067 Hex for parameter 103                   |
| 24 | // Subindex             |      |              |              |  |
| 25 | DB20.DBW 14             | HEX  | W#16#0000    | W#16#0000    |  |
| 26 | // Attribute            |      |              |              |  |
| 27 | DB20.DBB 16             | HEX  | B#16#10      | B#16#10      |  |
| 28 | // Number of elements   |      |              |              |  |
| 29 | DB20.DBB 17             | HEX  | B#16#01      | B#16#01      |  |
| 30 | // Parameter number     |      |              |              |  |
| 31 | DB20.DBW 18             | HEX  | W#16#0068    | W#16#0068    | 0068 Hex for parameter 104                   |
| 32 | // Subindex             |      |              |              |  |
| 33 | DB20.DBW 20             | HEX  |              | W#16#0000    | Response telegram                            |
| 35 | // Responce             |      |              |              |  |
| 36 | // Req. ref.            |      |              |              |  |
| 37 | DB21.DBB 0              | HEX  | B#16#03      |              |  |
| 38 | //Req ID                |      |              |              |  |
| 39 | DB21.DBB 1              | HEX  | B#16#01      |              |  |
| 40 | // No of Axis.          |      |              |              |  |
| 41 | DB21.DBB 2              | HEX  | B#16#01      |              |  |
| 42 | // Number of Par.       |      |              |              |  |
| 43 | DB21.DBB 3              | HEX  | B#16#03      |              |  |
| 44 | //Values                |      |              |              |  |
| 45 | DB21.DBW 4              | HEX  | W#16#0601    |              | 06 Hex = Datatype unsigned 16                |
| 46 | DB21.DBW 6              | HEX  | W#16#004B    |              | 01 Hex read date OK                          |
| 47 | DB21.DBW 8              | HEX  | W#16#0601    |              | 4B Hex = 75 dec corresponding to 0.75 kW.    |
| 48 | DB21.DBW 10             | HEX  | W#16#0190    |              | 190 Hex = 400 dec corresponding to 400 Volt. |
| 49 | DB21.DBW 12             | HEX  | W#16#0601    |              | 0601 Hex = 1500 dec corresponding to 50 Hz.  |
| 50 | DB21.DBW 14             | HEX  | W#16#0032    |              | 32 Hex = 50 dec corresponding to 50 Hz.      |
| 51 | DB21.DBW 16             | HEX  | W#16#0000    |              |  |



### ■ Multiple parameter write

This example shows a multiple write command to parameter 102 *Motor power*, parameter 207 *Ramp up time 1* and parameter 215 *Preset reference 1*.

The following values are written:

|   |         |
|---|---------|
| Parameter 102 <i>Motor power</i>        | 1.1 kW  |
| Parameter 207 <i>Ramp up time 1</i>     | 1.0 Sec |
| Parameter 215 <i>Preset reference 1</i> | 10.00 % |

Please also see the chapter *Description of DP V1 Read/Write Telegram* to see how the telegram is built up.

The screenshot shows a table with the following data:

| Address | Disp        | Status value            | Modify value |
|---------|-------------|-------------------------|--------------|
| 6       |             | // Axis                 |              |
| 7       | DB20.DBB 2  | HEX B#16#01             | B#16#01      |
| 8       |             | // Number of parameters |              |
| 9       | DB20.DBB 3  | HEX B#16#03             | B#16#03      |
| 10      |             | // Attribute            |              |
| 11      | DB20.DBB 4  | HEX B#16#10             | B#16#10      |
| 12      |             | // Number of elements   |              |
| 13      | DB20.DBB 5  | HEX B#16#01             | B#16#01      |
| 14      |             | // Parameter number     |              |
| 15      | DB20.DBW 6  | HEX W#16#0066           | W#16#0066    |
| 16      |             | // Subindex             |              |
| 17      | DB20.DBW 8  | HEX W#16#0000           | W#16#0000    |
| 18      |             | // Attribute            |              |
| 19      | DB20.DBB 10 | HEX B#16#10             | B#16#10      |
| 20      |             | // Number of elements   |              |
| 21      | DB20.DBB 11 | HEX B#16#01             | B#16#01      |
| 22      |             | // Parameter number     |              |
| 23      | DB20.DBW 12 | HEX W#16#00CF           | W#16#00CF    |
| 24      |             | // Subindex             |              |
| 25      | DB20.DBW 14 | HEX W#16#0000           | W#16#0000    |
| 26      |             | // Attribute            |              |
| 27      | DB20.DBB 16 | HEX B#16#10             | B#16#10      |
| 28      |             | // Number of elements   |              |
| 29      | DB20.DBB 17 | HEX B#16#01             | B#16#01      |
| 30      |             | // Parameter number     |              |
| 31      | DB20.DBW 18 | HEX W#16#00D7           | W#16#00D7    |
| 32      |             | // Subindex             |              |
| 33      | DB20.DBW 20 | HEX W#16#0000           | W#16#0000    |
| 34      |             | //Parameter values      |              |

Annotations on the right side of the screenshot:

- Request telegram (points to the entire table)
- 03 Hex for writing to three parameters (points to row 9)
- 10 Hex for a Value (points to row 11)
- 0066 Hex for parameter 102 (points to row 15)
- 00CF Hex for parameter 207 (points to row 23)
- 00D7 Hex for parameter 215 (points to row 31)

See the parameter values on the next page.

■ Multiple parameter write, cont.

|    | Address              | Disp | Status value | Modify value |  |
|----|----------------------|------|--------------|--------------|--|
| 34 | //Parameter values   |      |              |              |  |
| 35 | //Format             |      |              |              |  |
| 36 | DB20.DBB 22          | HEX  | B#16#06      | B#16#06      | 06 Hex is the word format unsigned 16 for parameter 102      |
| 37 | //Number of elements |      |              |              |  |
| 38 | DB20.DBB 23          | HEX  | B#16#01      | B#16#01      |  |
| 39 | //Value              |      |              |              |  |
| 40 | DB20.DBW 24          | HEX  | W#16#006E    | W#16#006E    | 6E Hex = 110 corresponding to 1.10 kW                        |
| 41 | //Format             |      |              |              |  |
| 42 | DB20.DBB 26          | HEX  | B#16#07      | B#16#07      | 07 Hex is the long word format unsigned 32 for parameter 207 |
| 43 | //Number of element  |      |              |              |  |
| 44 | DB20.DBB 27          | HEX  | B#16#01      | B#16#01      |  |
| 45 | //Value              |      |              |              |  |
| 46 | DB20.DBW 28          | HEX  | W#16#0000    | W#16#0000    |  |
| 47 | DB20.DBW 30          | HEX  | W#16#0064    | W#16#0064    | 64 Hex = 100 corresponding to 1.00 sec                       |
| 48 | //Format             |      |              |              |  |
| 49 | DB20.DBB 32          | HEX  | B#16#03      | B#16#03      | 03 Hex is the word format Interger 16 for parameter 215      |
| 50 | //Number of element  |      |              |              |  |
| 51 | DB20.DBB 33          | HEX  | B#16#01      | B#16#01      |  |
| 52 | //Value              |      |              |              |  |
| 53 | DB20.DBW 34          | HEX  | W#16#03E8    | W#16#03E8    | 3E8 Hex = 1000 corresponding to 10.00%                       |
| 54 |                      |      |              |              |  |
| 55 | // Responce          |      |              |              |  |
| 56 | // Req. ref.         |      |              |              |  |
| 57 | DB21.DBB 0           | HEX  | B#16#03      |              |  |
| 58 | //Req ID             |      |              |              |  |
| 59 | DB21.DBB 1           | HEX  | B#16#02      |              |  |
| 60 | // No of Axis.       |      |              |              |  |
| 61 | DB21.DBB 2           | HEX  | B#16#01      |              |  |
| 62 | // Number of Par.    |      |              |              |  |

DPV1\_PP08\_2\SIMATIC 300(1)\...\S7 Program(1)

**Extended Diagnose**

If a warning or alarm should occur on a VLT frequency converter it is possible to automatic give the Master a diagnose telegram that contains information about the warning or alarm.

In the VLT frequency converter the following parameter are mapped into the diagnose telegram:

- Parameter 540 *Warning word* (byte 10-13)
- Parameter 541 *Extended status word* (byte 14-17)
- Parameter 538 *Alarm word* (byte 18-21)
- Parameter 953 *Com. warning word* (byte 22-23)

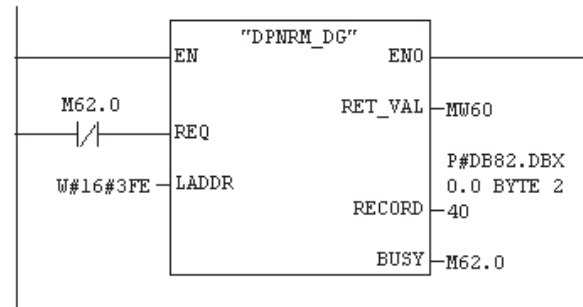
In parameter 849 *Extended diagnose* it is possible to select between the following:

- Disable* [0]: Standard diagnose will be sent. This choice is compatible with DPV0.
- Alarms* [1]: An alarm (parameter 538 *Alarm word*) or an event in parameter 953 *Com. warning word* will trigger a diagnose telegram and the extended diagnose telegram will be send to the master.
- Alarms and warnings* [2]: An alarm (parameter 538 *Alarm word*) or an warning (parameter 540 *Warning word*) or a event in parameter 953 *Com. warning word* will trigger a diagnose telegram and the extended diagnose telegram will be send to the master.

In S7 Manager the Diagnose address is set up in the Hardware and by Properties of the slave.  
In this example the Diagnostic address is set to 1022 dec.

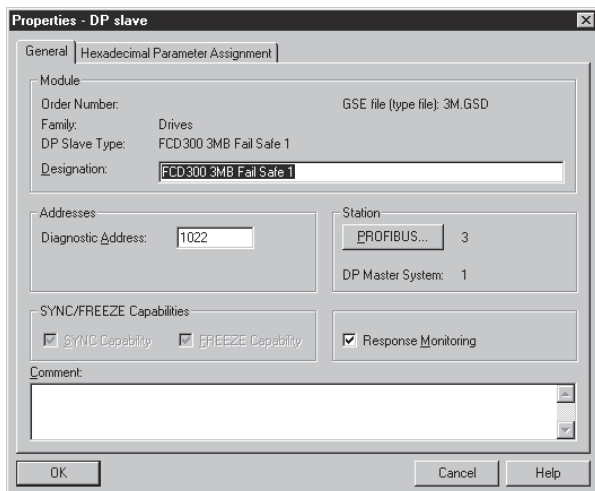
To read the diagnostic data SFC 13 "DPNRM\_DG" in Siemens is used. This Special Function Call is placed in OB 82 which will be called once if the extended diagnose is active due to an alarm or warning.

SFC 13 "DPNRM\_DG" will then read the diagnostic address 1022 (LADDR = 3FE Hex) and copy the data into Datablock DB82.



See examples on the next page.

---



■ **Extended Diagnose**

In this example the VLT frequency converter have a Live zero error, i.e. the current signal on terminal 60 is lower than parameter 315 *Term. 60 minimum scaling*. This warning will according the warning table give a live zero error of 10000 Hex. Parameter 849 *Extended diagnose* is programmed to *Alarms and warnings* [2].

The returned value 010000 Hex in Diagnose\_data [11] indicates a live zero error.

LAD/STL/FBD - [@DB82 -- DPV1\_PP08\_2\SIMATIC 300(1)\CPU 315-2 DP]

File Edit Insert PLC Debug View Options Window Help

| Address | Name              | Type | Initial value | Actual value |
|---------|-------------------|------|---------------|--------------|
| 0.0     | Diagnose_data[0]  | BYTE | B#16#0        | B#16#08      |
| 1.0     | Diagnose_data[1]  | BYTE | B#16#0        | B#16#0C      |
| 2.0     | Diagnose_data[2]  | BYTE | B#16#0        | B#16#00      |
| 3.0     | Diagnose_data[3]  | BYTE | B#16#0        | B#16#28      |
| 4.0     | Diagnose_data[4]  | BYTE | B#16#0        | B#16#04      |
| 5.0     | Diagnose_data[5]  | BYTE | B#16#0        | B#16#06      |
| 6.0     | Diagnose_data[6]  | BYTE | B#16#0        | B#16#12      |
| 7.0     | Diagnose_data[7]  | BYTE | B#16#0        | B#16#81      |
| 8.0     | Diagnose_data[8]  | BYTE | B#16#0        | B#16#00      |
| 9.0     | Diagnose_data[9]  | BYTE | B#16#0        | B#16#00      |
| 10.0    | Diagnose_data[10] | BYTE | B#16#0        | B#16#00      |
| 11.0    | Diagnose_data[11] | BYTE | B#16#0        | B#16#01      |
| 12.0    | Diagnose_data[12] | BYTE | B#16#0        | B#16#00      |
| 13.0    | Diagnose_data[13] | BYTE | B#16#0        | B#16#00      |
| 14.0    | Diagnose_data[14] | BYTE | B#16#0        | B#16#00      |
| 15.0    | Diagnose_data[15] | BYTE | B#16#0        | B#16#00      |
| 16.0    | Diagnose_data[16] | BYTE | B#16#0        | B#16#00      |
| 17.0    | Diagnose_data[17] | BYTE | B#16#0        | B#16#00      |
| 18.0    | Diagnose_data[18] | BYTE | B#16#0        | B#16#00      |
| 19.0    | Diagnose_data[19] | BYTE | B#16#0        | B#16#00      |
| 20.0    | Diagnose_data[20] | BYTE | B#16#0        | B#16#00      |
| 21.0    | Diagnose_data[21] | BYTE | B#16#0        | B#16#00      |

Press F1 to get Help.

28 Hex = Master address  
 0406 Hex = FCD 300 3 MB ident number  
 12 Hex = length of the block  
 81 Hex = Status message  
 82 Hex = Module status  
 Par. 540 Warning Byte 3  
 Par. 540 Warning Byte 2  
 Par. 540 Warning Byte 1  
 Par. 540 Warning Byte 0  
 Par. 541 Status Byte 3  
 Par. 541 Status Byte 2

■ **Extended Diagnose**

In this example the VLT frequency converter is current limit. This warning will according the warning table give a live zero error of 40 Hex.

Parameter 849 *Extended diagnose* is programmed to *Alarms and warnings* [2].

The returned value 40 Hex in Diagnose\_data [13] indicates a current limit.

| Address | Name              | Type | Initial value | Actual value |
|---------|-------------------|------|---------------|--------------|
| 0.0     | Diagnose_data[0]  | BYTE | B#16#0        | B#16#00      |
| 1.0     | Diagnose_data[1]  | BYTE | B#16#0        | B#16#0C      |
| 2.0     | Diagnose_data[2]  | BYTE | B#16#0        | B#16#00      |
| 3.0     | Diagnose_data[3]  | BYTE | B#16#0        | B#16#28      |
| 4.0     | Diagnose_data[4]  | BYTE | B#16#0        | B#16#04      |
| 5.0     | Diagnose_data[5]  | BYTE | B#16#0        | B#16#06      |
| 6.0     | Diagnose_data[6]  | BYTE | B#16#0        | B#16#12      |
| 7.0     | Diagnose_data[7]  | BYTE | B#16#0        | B#16#81      |
| 8.0     | Diagnose_data[8]  | BYTE | B#16#0        | B#16#00      |
| 9.0     | Diagnose_data[9]  | BYTE | B#16#0        | B#16#00      |
| 10.0    | Diagnose_data[10] | BYTE | B#16#0        | B#16#00      |
| 11.0    | Diagnose_data[11] | BYTE | B#16#0        | B#16#00      |
| 12.0    | Diagnose_data[12] | BYTE | B#16#0        | B#16#00      |
| 13.0    | Diagnose_data[13] | BYTE | B#16#0        | B#16#40      |
| 14.0    | Diagnose_data[14] | BYTE | B#16#0        | B#16#00      |
| 15.0    | Diagnose_data[15] | BYTE | B#16#0        | B#16#00      |
| 16.0    | Diagnose_data[16] | BYTE | B#16#0        | B#16#00      |
| 17.0    | Diagnose_data[17] | BYTE | B#16#0        | B#16#00      |
| 18.0    | Diagnose_data[18] | BYTE | B#16#0        | B#16#00      |
| 19.0    | Diagnose_data[19] | BYTE | B#16#0        | B#16#00      |
| 20.0    | Diagnose_data[20] | BYTE | B#16#0        | B#16#00      |
| 21.0    | Diagnose_data[21] | BYTE | B#16#0        | B#16#00      |

28 Hex = Master address  
 0406 Hex = FCD 300 3 MB ident number  
 12 Hex = length of the block  
 81 Hex = Status message  
 82 Hex = Module status  
 Par. 540 Warning Byte 3  
 Par. 540 Warning Byte 2  
 Par. 540 Warning Byte 1  
 Par. 540 Warning Byte 0  
 Par. 541 Status Byte 3  
 Par. 541 Status Byte 2

**■ Examples of DPV1 Parameter access services**

This example shows a single read command of parameter 207 Ramp up time 1. The ramp time is 3.00 sec.

The following examples show some of the parameter access services for the Profidrive Parameter Channel.

|                |                        |                       |
|----------------|------------------------|-----------------------|
| Request header | Request reference = 01 | Request id = 01       |
|                | Axis = 00              | Amount Parameter = 01 |
| Par number     | Attribute = 10         | Amount Elements = 01  |
|                | Par number = 00 CF     |                       |
|                | Subindex = 00 00       |                       |

Byte 0 Request reference: 01 Hex the master normally handles the Request reference.  
 Byte 1 Request id: 01 Hex for a read command (get par value)  
 Byte 2 Axis: Always 00 Hex  
 Byte 3 Amount parameter: 01 Hex as it is a single parameter read.  
 Byte 4 Attribute: 10 Hex for a read or write parameter values.  
 Byte 5 Amount elements: 01 Hex  
 Byte 6-7 Par. Numbers: 00 CF Hex for parameter 207  
 Byte 8-9 Sub index: 00 00 Hex. Par. 207 doesn't consist of sub-indexes.

reply positive

|                |                             |                       |
|----------------|-----------------------------|-----------------------|
| Reply-Header   | Reply reference mirror = 01 | Reply id = 01         |
|                | Axis 00                     | Amount Parameter = 01 |
| Parametervalue | Format = 07                 | Amount of values = 01 |
|                | Value = 00 00               |                       |
|                | Value = 01 2C               |                       |

Byte 0 Reply reference mirror: The Reply reference mirror is handle by the master.  
 Byte 1 Request id: 01 Hex means a positive read request  
 Byte 2 Axis: Always 00 Hex  
 Byte 3 Amount parameter: 01 Hex as it is a single parameter read.  
 Byte 4 Format: 07 Hex means that parameter 207 data type is Unsigned 32.  
 Byte 5 Amount of values: 01 Hex  
 Byte 6-9 Value: 00 00 01 2C Hex for 300 corresponding to 3.00 sec.

This example shows a single write command to parameter 207 Ramp up time 1. The ramp time is change to 10.00 sec.

request:

|                 |                        |                       |
|-----------------|------------------------|-----------------------|
| Request header  | Request reference = 01 | Request id = 02       |
|                 | Axis = 0               | Amount Parameter = 01 |
| Par number      | Attribute = 10         | Amount Elements = 01  |
|                 | Par number = 00 CF     |                       |
|                 | Subindex = 00 00       |                       |
| Parameter value | Format = 07            | Amount Values= 01     |
|                 | Value = 00 00 03 E8    |                       |

Byte 0 Request reference: 01 Hex the master normally handles the Request reference.  
 Byte 1 Request id: 02 Hex for a write command (change par value)  
 Byte 2 Axis: Always 00 Hex  
 Byte 3 Amount parameter: 01 Hex as it is a single parameter read.  
 Byte 4 Attribute: 10 Hex for a read or write parameter values.  
 Byte 5 Amount elements: 01 Hex  
 Byte 6-7 Par. Numbers: 00 CF Hex for parameter 207  
 Byte 8-9 Sub index: 00 00 Hex. Par. 207 doesn't consist of sub-indexes.  
 Byte 10 Format: 07 Hex as parameter 207 is a Unsigned 32.  
 Byte 11 Amount of values: 01 Hex  
 Byte 12-14 Value: 00 00 03 E8 Hex for 1000 corresponding to 10.00 sec.

reply positiv:

|                |                        |                       |
|----------------|------------------------|-----------------------|
| Request header | Request reference = 01 | Request id = 01       |
|                | Axis = 00              | Amount Parameter = 01 |

Byte 0 Reply reference mirror: The Reply reference mirror is handle by the master.  
 Byte 1 Request id: 02 Hex means a positive change parameter request  
 Byte 2 Axis: Always 00 Hex  
 Byte 3 Amount parameter: 01 Hex as it is a single parameter writes.

If the format of parameter 207 was wrong set the reply will be negative. In this example the format is set to an Unsigned 16 (06) instead of Unsigned 32 (07).

reply negative:

|                |                             |                       |
|----------------|-----------------------------|-----------------------|
| Reply-Header   | Reply reference mirror = 01 | Reply id = 82         |
|                | Axis mirror = 00            | Amount Parameter = 01 |
| Parametervalue | Format = 44                 | Amount of values = 01 |
|                | Error value = 00 05         |                       |

Byte 0 Reply reference mirror: The Reply reference mirror is handle by the master.  
 Byte 1 Request id: 82 Hex means a negative change parameter request  
 Byte 2 Axis: Always 00 Hex  
 Byte 3 Amount parameter: 01 Hex as it is a single parameter writes.  
 Byte 4 Format: 44 Hex means error.  
 Byte 5 Amount of values: 01 Hex  
 Byte 6-7 Value: 00 05 Hex means wrong datatype.

This example shows a single write command to index 3 in parameter 916 PCD configuration read. Value 520 is written to index 3.



**NOTE:** Note that sub-indexes starts in the VLT with index 1 and the Profibus DPV1 starts with index 0 i.e. that a VLT sub-index 1 is equal to DPV1 index 0.

|                 |                        |                       |
|-----------------|------------------------|-----------------------|
| Request header  | Request reference = 01 | Request id = 02       |
|                 | Axis = 0               | Amount Parameter = 01 |
| Par number      | Attribute = 10         | Amount Elements = 01  |
|                 | Par number = 03 94     |                       |
|                 | Subindex = 00 02       |                       |
| Parameter value | Format = 06            | Amount Values= 01     |
|                 | Value = 02 08          |                       |

Byte 0 Request reference: 01 Hex the master normally handles the Request reference.  
 Byte 1 Request id: 02 Hex for a write command (change par value)  
 Byte 2 Axis: Always 00 Hex  
 Byte 3 Amount parameter: 01 Hex as it is a single parameter writes.  
 Byte 4 Attribute: 10 Hex for a read or write parameter values.  
 Byte 5 Amount elements: 01 Hex  
 Byte 6-7 Par. Numbers: 03 94 Hex for parameter 916  
 Byte 8-9 Sub index: 00 02 Hex for writing to sub index 3 in par. 916.  
 Byte 10 Format: 06 Hex as parameter 916 is a Unsigned 16.  
 Byte 11 Amount of values: 01 Hex  
 Byte 12-14 Value: 02 08 Hex for 520



reply positive

|              |                             |                       |
|--------------|-----------------------------|-----------------------|
| Reply-Header | Reply reference mirror = 01 | Reply id = 02         |
|              | Axis mirror = 00            | Amount Parameter = 01 |

Byte 0 Reply reference mirror: The Reply reference mirror is handle by the master.  
 Byte 1 Request id: 02 Hex means a positive change parameter request  
 Byte 2 Axis: Always 00 Hex  
 Byte 3 Amount parameter: 01 Hex as it is a single parameter writes.

If the index number was wrong set the reply will be negative. In this example sub-index 10 is written which is not a valid index for parameter 916.

reply negative:

|                |                             |                       |
|----------------|-----------------------------|-----------------------|
| Reply-Header   | Reply reference mirror = 01 | Reply id = 82         |
|                | Axis mirror = 00            | Amount Parameter = 01 |
| Parametervalue | Format = 44                 | Amount of values =01  |
|                | Error value = 00 03         |                       |

Byte 0 Reply reference mirror: The Reply reference mirror is handle by the master.  
 Byte 1 Request id: 82 Hex means a negative change parameter request  
 Byte 2 Axis: Always 00 Hex  
 Byte 3 Amount parameter: 01 Hex as it is a single parameter writes.  
 Byte 4 Format: 44 Hex means error.  
 Byte 5 Amount of values: 01 Hex  
 Byte 6-7 Value: 00 03 Hex means wrong subindex.

---

This example shows a multiple read telegram of parameters 102-104.

Parameter 102 Motor power = 00.75 kW

Parameter 103 Motor voltage = 400 V

Parameter 104 Motor frequency = 50 Hz

request:

|                |                                   |                       |
|----------------|-----------------------------------|-----------------------|
| Request header | Request reference = 01            | Request id = 01       |
|                | Axis = 00                         | Amount Parameter = 03 |
| 1. Par number  | Attribute = 10                    | 1. Par number         |
|                | Par number = 00 66 Hex (par. 102) |                       |
|                | Subindex = 00 00                  |                       |
| 2. Par number  | Attribute = 10                    | 2. Par number         |
|                | Par number = 00 67 Hex (par. 103) |                       |
|                | Subindex = 00 00                  |                       |
| 3. Par number  | Attribute = 10                    | 3. Par number         |
|                | Par number = 00 68 (par. 104)     |                       |
|                | Subindex = 00 00                  |                       |

Byte 3 Amount parameter: 03 Hex, as this telegrams should read 3 parameters.



reply positive (complete):

|                   |                              |                       |
|-------------------|------------------------------|-----------------------|
| Reply-Header      | Reply reference mirror = 01  | Reply id = 01         |
|                   | Axis mirror = 00             | Amount Parameter = 03 |
| 1. Parametervalue | Format = 06                  | Amount of values = 01 |
|                   | Value = 00 4B Hex (0.75 kW)  |                       |
| 2. Parametervalue | Format = 06                  | Amount of values = 01 |
|                   | Value = 01 90 Hex (400 Volt) |                       |
| 3. Parametervalue | Format = 06                  | Amount of values = 01 |
|                   | Value = 32 Hex (50 Hz)       |                       |

---

This example shows a multiple write telegram to

par. 102-104. The parameters are change to:

Parameter 102 Motor power = 01.10 kW

Parameter 103 Motor voltage = 380 V

Parameter 104 Motor frequency = 60 Hz

request:

|                   |                                   |                       |
|-------------------|-----------------------------------|-----------------------|
| Request header    | Request reference = 01            | Request id = 02       |
|                   | Axis = 00                         | Amount Parameter = 03 |
| 1. Par number     | Attribute = 10                    | Amount Elements = 01  |
|                   | Par number = 00 66 Hex (par. 102) |                       |
|                   | Subindex = 00 00                  |                       |
| 2. Par number     | Attribute = 10                    | Amount Elements = 01  |
|                   | Par number = 00 67 Hex (par. 103) |                       |
|                   | Subindex = 00 00                  |                       |
| 3. Par number     | Attribute = 10                    | Amount Elements = 01  |
|                   | Par number = 00 68 (par. 104)     |                       |
|                   | Subindex = 00 00                  |                       |
| 1. Parametervalue | Format = 06                       | Amount of values = 01 |
|                   | Value = 00 6E Hex (110)           |                       |
| 2. Parametervalue | Format = 06                       | Amount of values = 01 |
|                   | Value = 01 7C (380)               |                       |
| 3. Parametervalue | Format = 06                       | Amount of values = 01 |
|                   | Value = 00 3C (60)                |                       |

Byte 3 Amount parameter: 03 Hex to write to three parameters.

reply positive:

|              |                             |                       |
|--------------|-----------------------------|-----------------------|
| Reply-Header | Reply reference mirror = 01 | Reply id = 02         |
|              | Axis mirror = 00            | Amount Parameter = 03 |

This example shows a single read telegram that reads the name of parameter 001 Language.

request:

|                |                        |                       |
|----------------|------------------------|-----------------------|
| Request header | Request reference = 01 | Request id = 01       |
|                | Axis = 00              | Amount Parameter = 01 |
| Par number     | Attribute = 20H        | Amount Elements = 1   |
|                | Par number = 00 01     |                       |
|                | Subindex = 06          |                       |

Byte 4 Attribute: 20 Hex for read parameter descriptions.  
 Byte 8-9 Sub index: 06 Hex for read the name of a parameter.

reply positive with text:

|                 |                             |                       |
|-----------------|-----------------------------|-----------------------|
| Reply-Header    | Reply reference mirror = 01 | Reply id = 01H        |
|                 | Axis mirror = 00            | Amount Parameter = 01 |
| Parameter value | Format = 09                 | Amount of values = 10 |
|                 | Byte 1 = 4C (L)             | Byte 2 = 41 (A)       |
|                 | Byte 3 = 4E (N)             | Byte 4 = 47 (G)       |
|                 | Byte 5 = 55 (U)             | Byte 6 = 41 (A)       |
|                 | Byte 7 = 47 (G)             | Byte 8 = 45 (E)       |

---

This example shows a single read telegram that reads index number [3] DANSK of parameter 001 Language.

request

|                |                        |                         |
|----------------|------------------------|-------------------------|
| Request header | Request reference = 01 | Request id = 01         |
|                | Axis = 00              | Amount Parameter = 01   |
| Par number     | Attribute = 30         | Amount of Elements = 01 |
|                | Par number = 00 01     |                         |
|                | Subindex = 00 03       |                         |

Byte 4 Attribute: 30 Hex for read parameter text.  
 Byte 8-9 Sub index: 3 Hex for read the text in array number 3.

reply positive:

|                 |                             |                       |
|-----------------|-----------------------------|-----------------------|
| Reply-Header    | Reply reference mirror = 01 | Reply id = 01         |
|                 | Axis mirror = 00            | Amount Parameter = 01 |
| Parameter value | Format = 09                 | Amount of values = 10 |
|                 | Byte 1 = 44 (D)             | Byte 2 = 41 (A)       |
|                 | Byte 3 = 4E (N)             | Byte 4 = 53 (S)       |
|                 | Byte 5 = 4b (K)             | Byte 6                |
|                 | Byte 7                      | Byte 8                |
|                 | Byte 9                      | Byte 10               |
|                 | Byte 11                     | Byte 12               |
|                 | Byte 13                     | Byte 14               |
|                 | Byte 15                     | Byte 16               |

**■ Example of DP V1 Read Parameter Service**

This example gives a detailed description of the telegrams and data needed for a Read of the value of VLT parameter 01 (Language).

The data bytes 0 through 3 controls the Profibus DP V1 Read / Write services including V1 error messages..

The data bytes 3 through 13 controls the specific Read of the VLT parameter 01.

**1: Write request service 5FH:**

|             |            |      |       |                     |           |                |      |              |                |              |               |              |               |              |
|-------------|------------|------|-------|---------------------|-----------|----------------|------|--------------|----------------|--------------|---------------|--------------|---------------|--------------|
| Byte        | 0          | 1    | 2     | 3                   | 4         | 5              | 6    | 7            | 8              | 9            | 10            | 11           | 12            | 13           |
| Value (Hex) | 5F         | 00   | 2F    | 0A                  | BB        | 01             | 00   | 01           | 10             | 01           | 00            | 01           | 00            | 00           |
| Meaning     | Funct. no. | slot | index | length              | Req. ref. | Req. ID = Read | Axis | No. of para. | Attri. = Value | No. of elem. | Par. No. High | Par. No. low | Sub. No. high | Sub. No. low |
| DP          | V1         |      |       | Drive profile V 3.0 |           |                |      |              |                |              |               |              |               |              |

2: Write response service 5FH: means request successfully received by the slave.

|         |            |      |       |        |
|---------|------------|------|-------|--------|
| Byte    | 0          | 1    | 2     | 3      |
| Value   | 5F         | 00   | 2F    | 00     |
| Meaning | Funct. no. | slot | index | length |
| DP      | V1         |      |       |        |

4: Read response - service DEH: Means the data is not ready yet.

|         |            |            |            |    |
|---------|------------|------------|------------|----|
| Byte    | 0          | 1          | 2          | 3  |
| Value   | DE         | 80         | B5         | 00 |
| Meaning | Funct. no. | Error code | Error code |    |
| DP      | V1         |            |            |    |

3: Read request service 5EH: The master asks the slave if the requested data is ready.

|         |            |      |       |        |
|---------|------------|------|-------|--------|
| Byte    | 0          | 1    | 2     | 3      |
| Value   | 5E         | 00   | 2F    | 00     |
| Meaning | Funct. no. | slot | index | length |
| DP      | V1         |      |       |        |

5: Read response+ service 5EH: The data is returned. Must be encoded from the Drives Profile part.

|         |            |      |       |        |                     |         |      |              |                     |                |                  |                           |
|---------|------------|------|-------|--------|---------------------|---------|------|--------------|---------------------|----------------|------------------|---------------------------|
| Byte    | 0          | 1    | 2     | 3      | 4                   | 5       | 6    | 7            | 8                   | 9              | 10               | 11                        |
| Value   | 5E         | 00   | 2F    | 08     | BB                  | 01      | 00   | 01           | 5                   | 01             | 00               | 00                        |
| Meaning | Funct. no. | slot | index | length | Req. ref.           | Res. ID | Axis | No. of para. | Format (unsigned 8) | No. of values. | Value 0, English | Empty, for byte alignment |
| DP      | V1         |      |       |        | Drive profile V 3.0 |         |      |              |                     |                |                  |                           |

**■ DP V1 Identifications.**

The V1 functionalities require a GSD file supporting V1. Of compatibility reasons in general the V1 versions got the same DP ident number as the corresponding V0 version. This means, that a V1 unit can replace a V0 unit without changing the master configuration. The table shows the available GSD files for FCM 300/FCD300/VLT 2800.

GSD files are placed on <http://www.danfoss.com/drives>.

| <b>GSD File Name</b> | <b>Description</b>   | <b>Ident nr.</b>          | <b>GSD Revision</b> |
|----------------------|----------------------|---------------------------|---------------------|
| <b>FCM 300</b>       |                      |                           |                     |
| DA010403.GSD         | FCM 300 V0 3 Mbaud   | 0403H<br>(old version)    | 01                  |
| DA020403.GSD         | FCM 300 V0 3 Mbaud   | 0403H<br>(actual version) | 02                  |
| DA010408.GSD         | FCM 300 V0 12Mbaud   | 0403H<br>(old version)    | 01                  |
| DA020408.GSD         | FCM 300 V0 12Mbaud   | 0403H<br>(actual version) | 02                  |
| <b>GSD File Name</b> |                      |                           |                     |
| <b>FCD 300</b>       |                      |                           |                     |
| DA010406.GSD         | FCD 300 V0 3 Mbaud   | 0406H<br>(old version)    | 01                  |
| DA010407.GSD         | FCD 300 V0 12 Mbaud  | 0407H<br>(old version)    | 01                  |
| DA020406.GSD         | FCD 300 V0 3 Mbaud   | 0406H<br>(actual version) | 02                  |
| DA020407.GSD         | FCD 300 V0 12 Mbaud  | 0407H<br>(actual version) | 02                  |
| DA030406.GSD         | FCD 300 V1 3 Mbaud   | 0406H<br>(actual version) | 03                  |
| DA030407.GSD         | FCD 300 V1 12 Mbaud  | 0407H<br>(actual version) | 03                  |
| <b>GSD File Name</b> |                      |                           |                     |
| <b>VLT 2800</b>      |                      |                           |                     |
| DA010404.GSD         | VLT 2800 V0 3 Mbaud  | 0404H<br>(old version)    | 01                  |
| DA020404.GSD         | VLT 2800 V0 3 Mbaud  | 0404H<br>(actual version) | 02                  |
| DA010405.GSD         | VLT 2800 V0 12 Mbaud | 0405H<br>(old version)    | 01                  |
| DA020405.GSD         | VLT 2800 V0 12 Mbaud | 0405H                     | 02                  |

**■ Data types**

| <b>Coding</b> | <b>PB type</b>                         | <b>VLT type</b> | <b>comment</b>   |
|---------------|--|-----------------|------------------|
| 1             | BOOL                                   | BOOLEAN         | standard type    |
| 2             | INTEGER8                               | SIGNED8         | standard type    |
| 3             | INTEGER16                              | SIGNED16        | standard type    |
| 4             | INTEGER32                              | SIGNED32        | standard type    |
| 5             | UNSIGNED8                              | UNSIGNED8       | standard type    |
| 6             | UNSIGNED16                             | UNSIGNED16      | standard type    |
| 7             | UNSIGNED32                             | UNSIGNED32      | standard type    |
| 8             | FLOAT                                  | FLOAT           | standard type    |
| 9             | VISIBLE STRING                         | VISIBLE STRING  | standard type    |
| 10            | OCTET STRING                           | OCTET STRING    | standard type    |
| 11            | TIMEOFDAY                              | -               | standard type    |
| 12            | TIMEOFDAY with date indication         | -               | standard type    |
| 13            | TIMEDIFFERENCE                         | -               | standard type    |
| 33            | N2                                     | -               | profile specific |
| 34            | N4                                     | -               | profile specific |
| 35            | V2 (bit sequence)                      | V2_TYPE (new)   | profile specific |
| 36            | L2                                     | -               | profile specific |
| 37            | R2                                     | -               | profile specific |
| 38            | T2                                     | -               | profile specific |
| 39            | T4                                     | -               | profile specific |
| 40            | NULL                                   | -               | profile specific |
| 41            | Byte                                   | -               | profile specific |
| 42            | Word                                   | -               | profile specific |
| 43            | double word                            | -               | profile specific |
| 44            | error                                  | -               | profile specific |
| 50            | DATE                                   | -               | standard type    |
| 52            | TIMEOFDAY without date indication      | -               | standard type    |
| 53            | TIMEDIFFERENCE with date indication    | -               | standard type    |
| 54            | TIMEDIFFERENCE without date indication | -               | standard type    |

**■ Size Attributes**

| physical variable | variable index | unit            | abbrev.           | conversion index |
|-------------------|----------------|-----------------|-------------------|------------------|
| Time              | 0              | none            |                   | 0                |
|                   | 4              | Sekunde         | s                 | 0                |
|                   |                | Minute          | min               | 70               |
|                   |                | Stunde          | h                 | 74               |
|                   |                | Tag             | d                 | 77               |
|                   |                | Millisekunde    | ms                | -3               |
|                   |                | Mikrosekunde    | ms                | -6               |
| Force             | 5              | Newton          | N                 | 0                |
|                   |                | Kilonewton      | kN                | 3                |
|                   |                | Meganewton      | MN                | 6                |
| Energy, Work      | 8              | Joule           | J                 | 0                |
|                   |                | Kilojoule       | kJ                | 3                |
|                   |                | Megajoule       | MJ                | 6                |
|                   |                | Wattstunde      | Wh                | 74               |
|                   |                | Kilowattstunde  | kWh               | 75               |
|                   |                | Megawattstunde  | MWh               | 76               |
| Effective Power   | 9              | Watt            | W                 | 0                |
|                   |                | Kilowatt        | kW                | 3                |
|                   |                | Megawatt        | MW                | 6                |
|                   |                | Milliwatt       | mW                | -3               |
| Apparent Power    | 10             | Voltampere      | VA                | 0                |
|                   |                | Kilovoltampere  | kVA               | 3                |
|                   |                | Megavoltampere  | MVA               | 6                |
|                   |                | Millivoltampere | mVA               | -3               |
| Speed             | 11             | 1/Sekunde       | s <sup>-1</sup>   | 0                |
|                   |                | 1/Minute        | min <sup>-1</sup> | 67               |
|                   |                | 1/Stunde        | h <sup>-1</sup>   | 72               |
| Torque            | 16             | Newtonmeter     | Nm                | 0                |
|                   |                | Kilonewtonmeter | kNm               | 3                |
|                   |                | Meganewtonmeter | MNm               | 6                |
| Temperature       | 17             | Kelvin          | K                 | 0                |
|                   |                | Grad Celsius    | °C                | 100              |
|                   |                | Grad Fahrenheit | °F                | 101              |
| Voltage           | 21             | V               | V                 |                  |
|                   |                | Volt            | kV                | 0                |
|                   |                | Kilovolt        | mV                | 3                |
|                   |                | Millivolt       | mV                | -3               |
| Current           | 22             | Mikrovolt       |                   | -6               |
|                   |                | Ampere          | A                 | 0                |
|                   |                | Milliampere     | mA                | -3               |
|                   |                | Kiloampere      | kA                | 3                |
| Resistance        | 23             | Mikroampere     | mA                | -6               |
|                   |                | Ohm             | W                 | 0                |
|                   |                | Milliohm        | mW                | -3               |
|                   |                | Kiloohm         | kW                | 3                |
| relative change   | 27             | Megaohm         | MW                | 6                |
|                   |                | Prozent         | %                 | 0                |
|                   |                | Frequency       |                   |                  |
|                   |                | Frequency       | 28                | Hertz            |
| Kilohertz         | kHz            |                 |                   | 3                |
| Megahertz         | Mhz            |                 |                   | 6                |
| Gigahertz         | GHz            |                 |                   | 9                |