OB1 - <offline>

""

Name: **Familie:**
Autor: **Version:** 0.1
Bausteinversion: 2
Zeitstempel Code: 12.06.2007 16:55:19
Interface: 15.02.1996 16:51:12
Längen (Baustein / Code / Daten): 02152 01888 00034

Name	Datentyp	Adresse	Kommentar
TEMP		0.0	
OB1_EV_C	Byte	0.0	Bits 0-3 = 1 (Coming event), Bits 4-7
OB1_SCAN	Byte	1.0	1 (Cold restart scan 1 of OB 1), 3 (Sc
OB1_PRIO	Byte	2.0	Priority of OB Execution
OB1_OB_N	Byte	3.0	1 (Organization block 1, OB1)
OB1_RESE	Byte	4.0	Reserved for system
OB1_RESE	Byte	5.0	Reserved for system
OB1_PREV	Int	6.0	Cycle time of previous OB1 scan (milli
OB1_MIN_	Int	8.0	Minimum cycle time of OB1 (millisecond
OB1_MAX_	Int	10.0	Maximum cycle time of OB1 (millisecond
OB1_DATE	Date_And_	12.0	Date and time OB1 started

Baustein: OB1 "Main Program Sweep (Cycle)"

```
Examples for inverter control by Profibus Master to ACTIVE/ACTIVE cube
with CM-PDPV1.

Network 1: Control of inverters start/stop and reference values
Network 2: parameter access by communication object PKW included
           in data-exchane objects
Network 3: parameter access by V1-channel


The system is build with one inverter equipped with CM-PDPV1 and three
additional inverters all connected together by CAN-Systembus.

All inverters are equipped with EM-moduls supporting CAN-Systembus.

Inverter 1 is equipped with CM-PDPV1 and is connected to the DP-Master.
Inverter 1 is the master of CAN-Systembus (CAN-Systembus NodeId = 0).
Inverter 2, 3, 4 are slaves on CAN-Systembus (CAN-Systembus NodeId = 1, 2, 3).
```

Netzwerk: 1 reference speed and control of inverters by data-exchange object

```
- start/stop control for each inverter
    inverters controlled by remote state machine (p.412 = 1)

- calculating the two reference speeds 1 and 2 for
    inverter 1 + 2 by analogue input 1
    inverter 3 + 4 by analogue input 2

    analogue input value is scaled to Profibus notation
    (0...100% ==> 0x0000...0x4000 / -100...0% = 0xC000...0)
    this value is referenced to Rated Frequency (p.375) or
    Profibus Reference (p.390) if p.390 is unequal zero (!)

    inverter 2, 3, 4 use Convert-Reference (p.1344) instead of
    Rated Frequeny (p.375) or Profibus Reference (p.390)

    the sign of the reference value can be changed individual for each inverter


  ***********************************************************************
  data exchange is proceeded by PEWxxx (process input) and
  PAWxxx (process output)as defined by the hardware configurator
  (PZD-objects in slot 2...5, I/O addresses)
  ***********************************************************************
```

```
// calculate speed reference 1 and store to inverter 1/2 positive/negative
// ***************************************************************************

      L     PEW  288    // load analogue input 1

      SRW   7           // scale to 0 ... 0x4000 = PDP notation
      L     105
      *I
      T     MW    20    // MW 20/22 = positive speed reference 1
      T     MW    22
```

```
      UN    E      0.1  // negate speed reference 1
      SPB   KR          // if S E0.1 is set
      L     MW    20
      L     -1
      *I
      T     MW    20

KR:   L     MW    20
      T     PAW  266    // store reference speed 1 to OUT-PZD2 (inverter 1)


      UN    E      0.3  // negate speed reference 1
      SPB   KR0         // if S E0.3 is set
      L     MW    22
      L     -1
      *I
      T     MW    22

KR0:  L     MW    22
      T     PAW  270    // store reference speed 1 to OUT-PZD4 (inverter 2)



// calculate speed reference 2 and store to inverter 3/4 positive/negative
// ***********************************************************************

      L     PEW  290    // load analogue input 2

      SRW   7           // scale to 0 ... 0x4000 = PDP notation
      L     105
      *I
      T     MW    20    // MW 20/22 = positive speed reference 2
      T     MW    22


      UN    E      0.5  // negate speed reference 2
      SPB   KR1         // if S E0.5 is set
      L     MW    20
      L     -1
      *I
      T     MW    20

KR1:  L     MW    20
      T     PAW  274    // store reference speed 2 to OUT-PZD6 (inverter 3)


      UN    E      0.7  // negate speed reference 2
      SPB   KR2         // if S E0.7 is set
      L     MW    22
      L     -1
      *I
      T     MW    22

KR2:  L     MW    22
      T     PAW  278    // store reference speed 2 to OUT-PZD8 (inverter 4)


// on/off control of inverter 1
// ****************************

      U     E      0.0  // switch on inverter 1, if S E0.0 is set
      SPBN  kr10

      L     W#16#F
      SPA   kr11

kr10: L     W#16#87     // switch off inverter 1, if S E0.0 is not set (plus error quit)

kr11: T     PAW  264    // store control word to OUT-PZD1


// on/off control of inverter 2
// ****************************

      U     E      0.2  // switch on inverter 2, if S E0.2 is set
      SPBN  kr12

      L     W#16#F
      SPA   kr13

kr12: L     W#16#87     // switch off inverter 1, if S E0.0 is not set (plus error quit)

kr13: T     PAW  268    // store control word to OUT-PZD3



// on/off control of inverter 3
// ****************************
```

```
      U     E      0.4  // switch on inverter 3, if S E0.2 is set
      SPBN  kr14

      L     W#16#F
      SPA   kr15

kr14: L     W#16#87     // switch off inverter 1, if S E0.0 is not set (plus error quit)

kr15: T     PAW  272    // store control word to OUT-PZD5


// on/off control of inverter 4
// ****************************

      U     E      0.6  // switch on inverter 4, if S E0.4 is set
      SPBN  kr16

      L     W#16#F
      SPA   kr17

kr16: L     W#16#87     // switch off inverter 1, if S E0.0 is not set (plus error quit)

kr17: T     PAW  276    // store control word to OUT-PZD3
```

Netzwerk: 2 usage of communication channel in data-exchange object

Parameters of inverters can be read/written by communication channel as defined
by the hardware configurator(PKW-object in slot 1, I/O addresses).

This data block is part of the DATA-EXCHANGE object, which is exchanged between
DP-Master and inverter cyclic. For this the data portion is transmitted on
EVERY data exchange cycle wether it is used or not.
The PKW-part of the predefined objects PPO1/PPO2 is named "4 Words PKW".
When building your own structure the object "PKW 4 Words IN/OUT" must be
selected and placed at slot 1. Only slot 1 is allowed for this object and the
object can be configured once only.

The PKW-data uses 8 bytes with consistency what causes the usage of SFC14/15
DP-Read/Write consistent data.

Data access is controlled by:
 - order/reply identification (PKE) word
 - Index   (IND) word
 - parameter value   (PWE) long

The upper byte of Index is assigned to the selected data set, the lower byte of
Index is assigned to the CAN-Systembus subaddress. Setting the lower byte of
Index to a value unequal to zero addresses the parameter access to a subscriber
on the CAN-Systembus.

Parameters of types int/uint/long can be accessed.
With the help of the index information (IND) the dataset of parameters is
selected and additional a CAN-Systembus subaddress can be set.

For detailled information see manual !!!

The following read/write cycles are implemented as an example for the usage of
the communication channel PKW:

- positive edge on E1.0 --> write Acceleration Clockwise to inverter 1
                            (p.420 dataset 5) 50,00 Hz/s (long value)

- negative edge on E1.0 --> write Acceleration Clockwise to inverter 1
                            (p.420 dataset 5) 5,00 Hz/s (long value)

- positive edge on E1.1 --> write Acceleration Clockwise to inverter 2
                            (p.420 dataset 5) 50,00 Hz/s (long value)

- negative edge on E1.1 --> write Acceleration Clockwise to inverter 2
                            (p.420 dataset 5) 5,00 Hz/s (long value)

- positive edge on E1.2 --> read RMS Current (p.211 dataset 0) from inverter 1
                                                               (uint value)
- negative edge on E1.2 --> read RMS Current (p.211 dataset 9) from inverter 2
                                                               (uint value)

MB100 is used to control read/write sequence (transfer-state)

--> MW40...47 = data record for SFC14 DPRD_DAT (8 bytes communication)
--> MW50...57 = data record for SFC15 DPWR_DAT (8 bytes communication)

Note: The read/write sequences can be monitored with the help of VPlus.
      See "Actual Values\Actual Frequ. Inv. Values\DP-Master OUT/IN"

```
            OUT = DP-Master to inverter
            IN  = inverter  to DP-Master
```

```
// ********************************
// STEP 1: set order identification
// ********************************

// positive edge on E1.0 ?
// -----------------------
      U     E      1.0
      FP    M      1.0
      SPBN  e1                  // no positive edge on E1.0


// at positive edge set order identification
// write Acceleration Clockwise to inverter 1 (p.420 dataset 5) 50,00 Hz/s

      L     W#16#8000           // order identification = 8 " write array long"
      L     420                 // p.420
      OW
      T     MW    50

      L     W#16#500            // IND = 0x0500 = dataset 5
      T     MW    52

      L     5000                // value = 5000 = 50,00 Hz
      T     MD    54

      L     1                   // state-transfer = 1 = "set request"
      T     MB   100

      SPA   e2


// negative edge on E1.0 ?
// -----------------------

e1:   U     E      1.0
      FN    M      1.1
      SPBN  e2                  // no negative edge on E1.0

// at negative edge set order identification
// write Acceleration Clockwise to inverter 1 (p.420 dataset 5)  5,00 Hz/s

      L     W#16#8000           // order identification = 8 " write array long"
      L     420                 // p.420
      OW
      T     MW    50

      L     W#16#500            // IND = 0x0500 = dataset 5
      T     MW    52

      L     500                 // value = 500 = 5,00 Hz
      T     MD    54

      L     1                   // state-transfer = 1 = "set request"
      T     MB   100


// positive edge on E1.1 ?
// -----------------------
e2:   U     E      1.1
      FP    M      1.2
      SPBN  e3                  // no positive edge on E1.1

// at positive edge set order identification
// write Acceleration Clockwise to inverter 2 (p.420 dataset 5) 50,00 Hz/s

      L     W#16#8000           // order identification = 8 " write array long"
      L     420                 // p.420
      OW
      T     MW    50

      L     W#16#501            // IND = 0x0501 = dataset 5 + CAN-Systembus address 1
      T     MW    52

      L     5000                // value = 5000 = 50,00 Hz
      T     MD    54

      L     1                   // state-transfer = 1 = "set request"
      T     MB   100

      SPA   e4


// negative edge on E1.1 ?
// -----------------------
e3:   U     E      1.1
      FN    M      1.3
      SPBN  e4                  // no negative edge on E1.1
```

```
// at negative edge set order identification
// write Acceleration Clockwise to inverter 2 (p.420 dataset 5)  5,00 Hz/s

      L     W#16#8000            // order identification = 8 " write array long"
      L     420                  // p.420
      OW
      T     MW    50

      L     W#16#501             // IND = 0x0501 = dataset 5 + CAN-Systembus address 1
      T     MW    52

      L     500                  // value = 500 = 5,00 Hz
      T     MD    54

      L     1                    // state-transfer = 1 = "set request"
      T     MB   100


// positive edge on E1.2 ?
// -----------------------
e4:   U     E      1.2
      FP    M      1.4
      SPBN  e5                   // no positive edge on E1.2

// at positive edge set order identification
// read RMS Current (p.211 dataset 0) from inverter 1

      L     W#16#1000            // order identification = 1 " read int/uint/long"
      L     211                  // p.211
      OW
      T     MW    50

      L     W#16#0               // IND = 0x0000 = dataset 0
      T     MW    52

      L     0                    // value = 0
      T     MD    54

      L     1                    // state-transfer = 1 = "set request"
      T     MB   100

      SPA   e6

// negative edge on E1.2 ?
// -----------------------
e5:   U     E      1.2
      FN    M      1.5
      SPBN  e6                   // no negative edge on E1.2

// at negative edge set order identification
// read RMS Current (p.211 dataset 0) from inverter 2
      L     W#16#1000            // order identification = 1 " read int/uint/long"
      L     211                  // p.211
      OW
      T     MW    50

      L     W#16#1               // IND = 0x0001 = dataset 0 + CAN-Systembus address 1
      T     MW    52

      L     0                    // value = 0
      T     MD    54

      L     1                    // state-transfer = 1 = "set request"
      T     MB   100


// ****************************************
// STEP 2: handling comunication by SFC14/15
// ****************************************

e6:   CALL  "DPRD_DAT"           // SFC14 "Read Consistent Data", read    SFC14              -- Read Consisten
                                 input data cyclic                       t Data of a Standard DP Slave
       LADDR  :=W#16#100         // 0x100 = 256 = start address of object (see hardware configuration)
       RET_VAL:=MW98             // return value of SFC14 (error code)
       RECORD :=P#M 40.0 BYTE 8  // data block (8 bytes)


// state-machine for communication cycles (see manual)
      L     MB   100             // load state-transfer
      SPL   END                  // list start
      SPA   IDLE                 // state-transfer=0, no action
      SPA   REQ                  // state-transfer=1, set request
      SPA   RSP                  // state-transfer=2, check response
END:  SPA   IDLE                 // list end


// set request
// -----------
REQ:  L     MW    40             // reply identification = 0?
```

```
      UW    W#16#F000
      SPN   IDLE                // jump if reply identification unequal zero

      CALL  "DPWR_DAT"          // SFC15 "Write Consistent Data", writ   SFC15              -- Write Consiste
                                e output data ONCE !!!                  nt Data to a Standard DP Slave
       LADDR  :=W#16#100        // 0x100 = 256 = start address of object (see hardware configuration)
       RECORD :=P#M 50.0 BYTE 8 // data block (8 bytes)
       RET_VAL:=MW96            // return value of SFC15 (error code)

      L     2                   // state-transfer = 2 = check response
      T     MB   100
      SPA   IDLE


// check response
// --------------
RSP:  L     MW    40            // reply identification = 0?
      UW    W#16#F000
      SPZ   IDLE                // jump if reply identification equal zero

// NOTE:
// In normal usage the reply identification must be checked here (see manual)
// For test the digital output 4.7 is toggled
      UN    A      4.7
      SPB   11

      R     A      4.7
      SPB   12

11:   S     A      4.7

12:   NOP   0

// clear output data to finish read/write-cycle
      L     0                   // clear order
      T     MW    50            // order identification
      T     MW    52            // IND
      T     MW    54            // value
      T     MW    56

      T     MB   100            // state-transfer = 0 = no action

      CALL  "DPWR_DAT"          // SFC15 "Write Consistent Data", writ   SFC15              -- Write Consiste
                                e output data ONCE !!!                  nt Data to a Standard DP Slave
       LADDR  :=W#16#100        // 0x100 = 256 = start address of object (see hardware configuration)
       RECORD :=P#M 50.0 BYTE 8 // data block (8 bytes)
       RET_VAL:=MW96            // return value of SFC15 (error code)


IDLE: NOP   0
```

Netzwerk: 3 usage of V1-channel

```
The V1-channel is a method for read/write data from/to an inverter without the
necessity of cyclic data-exchange. The bus load is reduced (no cyclic
data-exchange with communication-channel as described before).
Only in a situation where a data access is necesary the read/write V1 telegrams
are send on the bus.

All types of inverter parameters can be accessed (int/uint/long/string).

NOTE:
When using DP-V1 channel with S7 PLC the p.329 "DP-V1 Mode" must be set to "2 -
S7 compatible".
For DP-V1 read/write access SFC52/53 must be used.


The inverter is addressed by his Profibus diagnostic address (defined in
hardware configuration). The addressing itself uses:

slot  = 0: allways (defined by ID = diagnostic address)

index = 1: setting of parameter number, data set, CAN-Systembus subaddress
index = 2: read/write parameter value


Reading/Writing parameters needs 2 steps.

Step 1:
Write index 1 for Setting parameter number, data set and CAN-Systembus
subaddress

Step 2:
Read or write parameter values by index 2


The following read/write cycles are implemented as an example for the usage of
the DP-V1 chanel:
```

```
- positive edge on E1.3 --> write Acceleration Clockwise to inverter 1
                            (p.420 dataset 5) 50,00 Hz/s (long value)

- negative edge on E1.3 --> write Acceleration Clockwise to inverter 1
                            (p.420 dataset 5)  5,00 Hz/s (long value)


- positive edge on E1.4 --> write Acceleration Clockwise to inverter 2
                            (p.420 dataset 5) 50,00 Hz/s (long value)

- negative edge on E1.4 --> write Acceleration Clockwise to inverter 2
                            (p.420 dataset 5)  5,00 Hz/s (long value)


- positive edge on E1.5 --> write warning limit long term Ixt to inverter 1
                            (p.406 dataset 0) 50 % (uint value)

- negative edge on E1.5 --> write warning limit long term Ixt to inverter 2
                            (p.406 dataset 0) 50 % (uint value)


- positive edge on E1.6 --> read stator frequency from inverter 1
                            (p.210 dataset 0)  (long value)

- negative edge on E1.6 --> read stator frequency from inverter 2
                            (p.210 dataset 0)  (long value)


- positive edge on E1.7 --> read RMS Current (p.211 dataset 0) from inverter 1
                                                               (uint value)
- negative edge on E1.7 --> read RMS Current (p.211 dataset 0) from inverter 2
                                                               (uint value)

MB110 is used to control read/write sequence (transfer-state)
```

```
// positive edge on E1.3 ?
// -----------------------
      U     E      1.3
      FP    M      4.0

      SPBN  V1                                  // no positive edge on E1.3


// at positive edge set order
// write acceleration clockwise to inverter 1

      L     DW#16#7FE                           // diagnostic address inverter 1 / slot = 0
      T     MD   200                            // see hardware conf. adress space
      L     W#16#1                              // index =1
      T     MW   204
      L     4                                   // LEN = 4 bytes
      T     MW   206

      L     420                                 // p.420
      T     MW   220
      L     5                                   // data set 5
      T     MB   222
      L     0                                   // CAN-Systembus subaddress 0
      T     MB   223

      L     5000                                // value p.420 = 50,00 Hz/s
      T     MD   120
      L     4                                   // LEN = 2 for p.420 data
      T     MW   112

      L     1                                   // set write sequence step 1
      T     MB   110

      SPA   V20



// negative edge on E1.3 ?
// -----------------------
V1:   U     E      1.3
      FN    M      4.1

      SPBN  V2                                  // no negative edge on E1.3


// at negative edge set order
// write acceleration clockwise to inverter 1

      L     DW#16#7FE                           // diagnostic address inverter 1 / slot = 0
      T     MD   200                            // see hardware conf. adress space
      L     W#16#1                              // index =1
      T     MW   204
      L     4                                   // LEN = 4 bytes
```

```
      T     MW   206

      L     420                              // p.420
      T     MW   220
      L     5                                // data set 5
      T     MB   222
      L     0                                // CAN-Systembus subaddress 0
      T     MB   223

      L     500                              // value p.420 = 5,00Hz/s
      T     MD   120
      L     4                                // LEN = 4 for p.420 data
      T     MW   112

      L     1                                // set write sequence step 1
      T     MB   110

      SPA   V20


// positive edge on E1.4 ?
// -----------------------
V2:   U     E      1.4
      FP    M      4.2

      SPBN  V3                               // no positive edge on E1.4

// at positive edge set order
// write acceleration clockwise to inverter 2

      L     DW#16#7FE                        // diagnostic address inverter 1 / slot = 0
      T     MD   200                         // see hardware conf. adress space
      L     W#16#1                           // index =1
      T     MW   204
      L     4                                // LEN = 4 bytes
      T     MW   206

      L     420                              // p.420
      T     MW   220
      L     5                                // data set 5
      T     MB   222
      L     1                                // CAN-Systembus subaddress 1
      T     MB   223

      L     5000                             // value p.420 = 50,00Hz/s
      T     MD   120
      L     4                                // LEN = 4 for p.420 data
      T     MW   112

      L     1                                // set write sequence step 1
      T     MB   110

      SPA   V20



// negative edge on E1.4 ?
// -----------------------
V3:   U     E      1.4
      FN    M      4.3

      SPBN  V4                               // no negative edge on E1.6

// at negative edge set order
// write acceleration clockwise to inverter 2

      L     DW#16#7FE                        // diagnostic address inverter 1 / slot = 0
      T     MD   200                         // see hardware conf. adress space
      L     W#16#1                           // index =1
      T     MW   204
      L     4                                // LEN = 4 bytes
      T     MW   206

      L     420                              // p.420
      T     MW   220
      L     5                                // data set 5
      T     MB   222
      L     1                                // CAN-Systembus subaddress 1
      T     MB   223

      L     500                              // value p.420 = 0,00Hz
      T     MD   120
      L     4                                // LEN = 4 for p.480 data
      T     MW   112

      L     1                                // set write sequence step 1
      T     MB   110

      SPA   V20
```

```
// positive edge on E1.5 ?
// -----------------------
V4:   U     E      1.5
      FP    M      4.4

      SPBN  V5                              // no positive edge on E1.5

// at positive edge set order
// write warning limit long term Ixt to inverter 1

      L     DW#16#7FE                       // diagnostic address inverter 1 / slot = 0
      T     MD   200                        // see hardware conf. adress space
      L     W#16#1                          // index =1
      T     MW   204
      L     4                               // LEN = 4 bytes
      T     MW   206

      L     406                             // p.406
      T     MW   220
      L     0                               // data set 0
      T     MB   222
      L     0                               // CAN-Systembus subaddress 0
      T     MB   223

      L     50                              // value p.406 = 50%
      T     MW   120
      L     2                               // LEN = 2 for p.405 data
      T     MW   112

      L     1                               // set write sequence step 1
      T     MB   110



// negative edge on E1.5 ?
// -----------------------
V5:   U     E      1.5
      FN    M      4.5

      SPBN  V6                              // no negative edge on E1.5

// at negative edge set order
// write warning limit long term Ixt to inverter 2

      L     DW#16#7FE                       // diagnostic address inverter 1 / slot = 0
      T     MD   200                        // see hardware conf. adress space
      L     W#16#1                          // index =1
      T     MW   204
      L     4                               // LEN = 4 bytes
      T     MW   206

      L     406                             // p.406
      T     MW   220
      L     0                               // data set 0
      T     MB   222
      L     1                               // CAN-Systembus subaddress 1
      T     MB   223

      L     50                              // value p.406 = 50%
      T     MW   120
      L     2                               // LEN = 2 for p.405 data
      T     MW   112

      L     1                               // set write sequence step 1
      T     MB   110



// positive edge on E1.6 ?
// -----------------------
V6:   U     E      1.6
      FP    M      4.6

      SPBN  V7                              // no negative edge on E1.6

// at positive edge set order
// read stator frequency from inverter 1

      L     DW#16#7FE                       // diagnostic address inverter 1 / slot = 0
      T     MD   200                        // see hardware conf. adress space
      L     W#16#1                          // index =1
      T     MW   204
      L     4                               // LEN = 4 bytes
      T     MW   206

      L     210                             // p.210
      T     MW   220
      L     0                               // data set 0
      T     MB   222
```

```
      L     0                               // CAN-Systembus subaddress 0
      T     MB   223

      L     20                              // max. data length to read
      T     MW   208

      L     5                               // set read sequence step 1
      T     MB   110

      SPA   V20



// negative edge on E1.6 ?
// ----------------------
V7:   U     E      1.6
      FN    M      4.7

      SPBN  V8                              // no negative edge on E1.6

// at positive edge set order
// read stator frequency from inverter 2

      L     DW#16#7FE                       // diagnostic address inverter 1 / slot = 0
      T     MD   200                        // see hardware conf. adress space
      L     W#16#1                          // index =1
      T     MW   204
      L     4                               // LEN = 4 bytes
      T     MW   206

      L     210                             // p.210
      T     MW   220
      L     0                               // data set 0
      T     MB   222
      L     1                               // CAN-Systembus subaddress 1
      T     MB   223

      L     20                              // max. data length to read
      T     MW   208

      L     5                               // set read sequence step 1
      T     MB   110

      SPA   V20



// positive edge on E1.7 ?
// ----------------------
V8:   U     E      1.7
      FP    M      5.1

      SPBN  V9                              // no positive edge on E1.7

// at positive edge set order
// read RMS current from inverter 1

      L     DW#16#7FE                       // diagnostic address inverter 1 / slot = 0
      T     MD   200                        // see hardware conf. adress space
      L     W#16#1                          // index =1
      T     MW   204
      L     4                               // LEN = 4 bytes
      T     MW   206

      L     211                             // p.211
      T     MW   220
      L     0                               // data set 0
      T     MB   222
      L     0                               // CAN-Systembus subaddress 0
      T     MB   223

      L     20                              // max. data length to read
      T     MW   208

      L     5                               // set read sequence step 1
      T     MB   110

      SPA   V20


// negative edge on E1.7 ?
// ----------------------
V9:   U     E      1.7
      FN    M      5.2

      SPBN  V20                             // no negative edge on E1.7

// at negative edge set order
// read RMS current from inverter 2
```

```
      L     DW#16#7FE                       // diagnostic address inverter 1 / slot = 0
      T     MD   200                        // see hardware conf. adress space
      L     W#16#1                          // index =1
      T     MW   204
      L     4                               // LEN = 4 bytes
      T     MW   206

      L     211                             // p.211
      T     MW   220
      L     0                               // data set 0
      T     MB   222
      L     1                               // CAN-Systembus subaddress 1
      T     MB   223

      L     20                              // max. data length to read
      T     MW   208

      L     5                               // set read sequence step 1
      T     MB   110



// ---------------------------------------------------------------------
V20:  NOP   0

// state machine for DP-V1 communication cycles
      L     MB   110
      SPL   LST1                            // list start
      SPA   IDL1                            // state = 0: no request active
      SPA   V1W1                            // state = 1: write request step 1
      SPA   V1W2                            // state = 2: write request step 2
      SPA   V1W3                            // state = 3: write request step 3
      SPA   V1W4                            // state = 4: write request step 4
      SPA   V1R1                            // state = 5: read  request step 1
      SPA   V1R2                            // state = 6: read  request step 2
      SPA   V1R3                            // state = 7: read  request step 3
      SPA   V1R4                            // state = 8: read  request step 4
LST1: SPA   IDL1                            // list end



// write sequence step 1
V1W1: S     M      3.0                      // set REQUEST
      L     2                               // set state = 2, write request step 2
      T     MB   110
      SPA   V21                             // process SFC53 DP-V1-Write


// write sequence step 2
V1W2: UN    M      3.1                      // DONE = TRUE ?
      SPB   V21                             // process DP-V1-Write

      R     M      3.0                      // reset REQUEST
      R     M      3.1                      // reset DONE
      L     3                               // set state = 3, write request step 3
      T     MB   110
      SPA   V21                             // process SFC53 DP-V1-Write


// write sequence step 3
V1W3: S     M      3.0                      // set REQUEST
      L     W#16#2                          // index = 2
      T     MW   204
      L     MW   112                        // load parameter value length
      T     MW   206

      L     MW   112                        // 2/4 byte data ?
      L     W#16#2
      ==I
      SPBN  B4

      L     MW   120                        // load 2 bytes data
      T     MW   220
      SPA   Bx

B4:   L     MD   120                        // load 4 bytes data
      T     MD   220

Bx:   L     4                               // set state = 4, write request step 4
      T     MB   110
      SPA   V21                             // process DP-V1_Write


// write sequence step 4
V1W4: UN    M      3.1                      // Write DONE = TRUE ?
      SPB   V21                             // process DP-V1-Write

      R     M      3.0                      // reset REQUEST
      R     M      3.1                      // reset DONE
```

```
      L     0                               // set state = 0, no request active
      T     MB   110
      SPA   V21

// ---------------------------------------------------------------------

// read sequence step 1
V1R1: S     M      3.0                      // set REQUEST
      L     6                               // set state = 6, read request step 2
      T     MB   110
      SPA   V21                             // process SFC53 DP-V1-Write


// read sequence step 2
V1R2: UN    M      3.1                      // DONE = TRUE ?
      SPB   V21                             // process DP-V1-Write

      R     M      3.0                      // reset REQUEST
      R     M      3.1                      // reset DONE
      L     7                               // set state = 7, read request step 3
      T     MB   110
      SPA   V21                             // process SFC53 DP-V1-Write


// read sequence step 3
V1R3: S     M      3.4                      // set REQUEST
      L     W#16#2                          // index = 2
      T     MW   204
      L     8                               // set state = 8, read request step 4
      T     MB   110
      SPA   V22                             // process SFC52 DP-V1-Read


// read sequence step 4
V1R4: UN    M      3.5                      // Read VALID = TRUE ?
      SPB   V22                             // process DP-V1-Read

      R     M      3.4                      // reset REQUEST
      R     M      3.5                      // reset VALID

      L     0                               // set state = 0, no request active
      T     MB   110                        // process DP-V1-read
      SPA   V22


// ---------------------------------------------------------------------
V21:  NOP   0

      CALL  "WRREC" , "instance_db_for_sfb53"                                SFB53 / DB53       -- Write a
                                                                              Process Data Record
       REQ   :=M3.0                         // REQUEST
       ID    :=MD200                        // address
       INDEX :=MW204                        // index
       LEN   :=MW206                        // length of parameter data to write
       DONE  :=M3.1                         // DP-V1 write ready
       BUSY  :=M3.2                         // DP-V1 write access in progress
       ERROR :=M3.3                         // DP-V1 write with error reply
       STATUS:=MD210                        // DP-V1 write access status
       RECORD:=P#M 220.0 BYTE 24            // parameter data location

      SPA   IDL1


V22:  NOP   0

      CALL  "RDREC" , "instance_db_for_sfb52"                                SFB52 / DB52       -- Read a
                                                                             Process Data Record
       REQ   :=M3.4                         // REQUEST
       ID    :=MD200                        // address
       INDEX :=MW204                        // index
       MLEN  :=MW208                        // max. parameter data length
       VALID :=M3.5                         // DP-V1 read ready (data valid)
       BUSY  :=M3.6                         // DP-V1 read access in progress
       ERROR :=M3.7                         // DP-V1 read access with error reply
       STATUS:=MD212                        // DP-V1 read access status
       LEN   :=MW214                        // length of parameter data read
       RECORD:=P#M 240.0 BYTE 20            // parameter data location

// ---------------------------------------------------------------------
IDL1: NOP   0

// read out data from inverter(s)
      L     MD   240                        // 4 bytes data
      L     MW   240
```