

**OB1 - <offline>**

"Cycle Execution"

**Name:**  
**Autor:**  
**Zeitstempel Code:**  
**Interface:**  
**Längen (Baustein / Code / Daten):**

**Familie:**  
**Version:** 0.1  
**Bausteinversion:** 2  
14.07.2010 11:31:25  
06.01.2004 11:26:53  
00678 00558 00034

Name	Datentyp	Adresse	Kommentar
TEMP		0.0	
OB1_EV_CLASS	Byte	0.0	Bits 0-3 = 1 (Coming event), Bits 4-7 = 1 (Event class 1)
OB1_SCAN_1	Byte	1.0	1 (Cold restart scan 1 of OB 1), 3 (Scan 2-n of OB 1)
OB1_PRIORITY	Byte	2.0	Priority of OB Execution
OB1_OB_NUMBR	Byte	3.0	1 (Organization block 1, OB1)
OB1_RESERVED_1	Byte	4.0	Reserved for system
OB1_RESERVED_2	Byte	5.0	Reserved for system
OB1_PREV_CYCLE	Int	6.0	Cycle time of previous OB1 scan (milliseconds)
OB1_MIN_CYCLE	Int	8.0	Minimum cycle time of OB1 (milliseconds)
OB1_MAX_CYCLE	Int	10.0	Maximum cycle time of OB1 (milliseconds)
OB1_DATE_TIME	Date_And_Time	12.0	Date and time OB1 started

**Baustein: OB1 "Lesen/Schreiben MP85ADP"**

Standardfunktionen

Netzwerk: 1 Bruttomessert X lesen (mit Status)

Peripherie lesen nach MD50 mit SFC 14

```
CALL "DPRD_DAT"           //Aufruf SFC14                SFC14           -- Lese konsisten
                           //                           te Daten
LADDR :=W#16#100           //Lade Daten von HW Adresse 256dez -> 100hex
RET_VAL:=MW200             //Zieadresse Fehlerfall
RECORD :=P#M 50.0 BYTE 6   //Speichern der Daten in Merkerdoppelwort 50, 6 Byte
```

Netzwerk: 2 Bruttomessert Y lesen (mit Status)

Peripherie lesen nach MD60 mit SFC14

```
CALL "DPRD_DAT"           //Aufruf SFC14                SFC14           -- Lese konsisten
                           //                           te Daten
LADDR :=W#16#106           //Lade Daten von HW Adresse 262dez -> 106hex
RET_VAL:=MW201             //Zieadresse Fehlerfall
RECORD :=P#M 60.0 BYTE 6   //Speichern der Daten in Merkerdoppelwort 60, 6 Byte
```

Netzwerk: 3 Steuerwort schreiben

Schreiben in Peripherie von Merkerdoppelwort 70

```
CALL "DPWR_DAT"           //Call SFC15                  SFC15           -- Schreibe konsi
                           //                           stente Daten
LADDR :=W#16#100           //Schreibe Steuerwort zu Ausgangsadresse 256dez -> 100hex
RECORD :=P#M 70.0 BYTE 4   //Lade Bitmuster 32 Bit (Steuerwort 1 u. 2 ) vom Merkerdoppelwort, 4 byte
RET_VAL:=MW202             //Zieadresse Fehlerfall
```

Netzwerk: 4      Werkstück ID azyklisch lesen mit SFB52, DPV1
---

Lese Daten via DPV1
---------------------

```
CALL  "RDREC" , DB52                                SFB52          -- Read a Process
                                                    Data Record

REQ   :=M15.1      //Start Ausführung, Merker 15.1 setzen
ID    :=DW#16#10C   //Reale Adresse aus HW-Konfig des jeweiligen slots (Prozess-Satus 268dez -> 1
                    0C hex)
INDEX :=40          //Index des gewünschten Werts, Prozess_Stat = 28h -> 40dez = Werkstück ID les
                    en
MLEN  :=59          //Datenlänge, visible string 59 Zeichen = 59 byte
VALID :=DB52.DBX10.0
BUSY  :=DB52.DBX10.1
ERROR :=DB52.DBX10.2
STATUS:=DB52.DBD12
LEN   :=DB52.DBW16
RECORD:=P#M 80.0 BYTE 59 //Schreiben in das Merkerwort 80, 59 byte
```

Netzwerk: 5      Werkstück ID azyklisch schreiben mit SFB53, DPV1
---

Schreiben Daten via DPV1
--------------------------

```
CALL  "WRREC" , DB53                                SFB53          -- Write a Proces
                                                    s Data Record

REQ   :=M15.2      //Start Ausführung, Merker 15.2 setzen
ID    :=DW#16#10C   //Reale Adresse aus HW-Konfig des jeweiligen slots
INDEX :=127         //Index des gewünschten Werts, Prozess_Stat = 7Fh = Werkstück ID schreiben
LEN   :=59
DONE  :=DB53.DBX10.0
BUSY  :=DB53.DBX10.1
ERROR :=DB53.DBX10.2
STATUS:=DB53.DBD16
RECORD:=P#M 140.0 BYTE 59 //Schreiben in das Merkerwort 140, 59 byte
```

BE