

DATA_BLOCK DB 41
TITLE =
AUTHOR : SIMATIC
FAMILY : ICONT
VERSION : 0.0

FB 41
BEGIN

```
COM_RST := FALSE;
MAN_ON := TRUE;
PVPER_ON := FALSE;
P_SEL := TRUE;
I_SEL := TRUE;
INT_HOLD := FALSE;
I_ITL_ON := FALSE;
D_SEL := FALSE;
CYCLE := T#1S;
SP_INT := 0.000000e+000;
PV_IN := 0.000000e+000;
PV_PER := W#16#0;
MAN := 0.000000e+000;
GAIN := 2.000000e+000;
TI := T#20S;
TD := T#10S;
TM_LAG := T#2S;
DEADB_W := 0.000000e+000;
LMN_HLM := 1.000000e+002;
LMN_LLM := 0.000000e+000;
PV_FAC := 1.000000e+000;
PV_OFF := 0.000000e+000;
LMN_FAC := 1.000000e+000;
LMN_OFF := 0.000000e+000;
I_ITLVAL := 0.000000e+000;
DISV := 0.000000e+000;
LMN := 0.000000e+000;
LMN_PER := W#16#0;
QLMN_HLM := FALSE;
QLMN_LLM := FALSE;
LMN_P := 0.000000e+000;
LMN_I := 0.000000e+000;
LMN_D := 0.000000e+000;
PV := 0.000000e+000;
ER := 0.000000e+000;
sInvAlt := 0.000000e+000;
sIanteilAlt := 0.000000e+000;
sRestInt := 0.000000e+000;
sRestDif := 0.000000e+000;
sRueck := 0.000000e+000;
sLmn := 0.000000e+000;
sbArwHLMon := FALSE;
sbArwLLMon := FALSE;
sbILimOn := TRUE;
```

END_DATA_BLOCK

ORGANIZATION_BLOCK OB 35
TITLE = "Cyclic Interrupt"
VERSION : 0.1

VAR_TEMP

```
OB35_EV_CLASS : BYTE ; //Bits 0-3 = 1 (Coming event), Bits 4-7 = 1 (Event class 1)
OB35_STRT_INF : BYTE ; //16#36 (OB 35 has started)
OB35_PRIORITY : BYTE ; //Priority of OB Execution
OB35_OB_NUMBR : BYTE ; //35 (Organization block 35, OB35)
OB35_RESERVED_1 : BYTE ; //Reserved for system
OB35_RESERVED_2 : BYTE ; //Reserved for system
OB35_PHASE_OFFSET : WORD ; //Phase offset (msec)
OB35_RESERVED_3 : INT ; //Reserved for system
OB35_EXC_FREQ : INT ; //Frequency of execution (msec)
```

```

OB35_DATE_TIME : DATE_AND_TIME ;      //Date and time OB35 started
END_VAR
BEGIN
NETWORK
TITLE =wenn switching from Auto->Hand then takeover output on Handvalue
//Flankdetection on Automatik to Manual
  A    M    10.0; // If hand is choossen then switch the actual output on the hand value
  FP   M    11.0; // one time !!, so a positive flank
  =    M    11.1;
  JCN  cont; // When no flank then to the controller

  L    MD   48; // Output
  T    MD   20; // --> Manual value
NETWORK
TITLE =Controller
//Not all the parameters are necessary on the FB, here this is done to test it
//easy in a VAT-table.
//
//The values from the Analog Input and the Analog Output are here 27648 bitvalues
//are 100%. When that is not wanted, the parameters "PV_FAC" for the Process
//value and "LMN_FAC" for the output must be changed.
//With the parameters "PV_OFF" and "LMN_OFF" the underlimit has to be changed
//wenn that is not zero.
//
cont: CALL FB    41 , DB    41 (
  COM_RST      := FALSE, // Total Reset
  MAN_ON       := M    10.0, // For switching to handvalue
  PVPER_ON     := TRUE, // Read in ProcesValue over Analog Input on "PV_VER"
  P_SEL        := TRUE, // P-Action ON
  I_SEL        := M    10.1, // I-Action ON
  INT_HOLD     := M    10.2, // The output of the integrator can be "frozen" by setting the input "integral action hold."
  I_ITL_ON     := FALSE, // INITIALIZATION OF THE INTEGRAL ACTION
  D_SEL        := M    10.3, // DERIVATIVE ACTION ON
  CYCLE        := T#1S, // The time between the block calls must be constant. The "sampling time" input specifies the time between block calls.
  SP_INT       := MD   12, // Setpoint
  PV_IN        := MD   16, // Is switched OFF with PVPER_ON = TRUE
  PV_PER       := PIW  512, // Value from Analog Input
  MAN          := MD   20, // The "manual value" input is used to set a manual value using the operator interface functions.
  GAIN         := MD   24,
  TI           := MD   28,
  TD           := MD   32,
  DEADB_W      := MD   36, // A dead band is applied to the error. The "dead band width" input determines the size of the dead band.
  LMN_HLM      := MD   40, // The manipulated value is always limited by an upper and lower limit.
  LMN_LLM      := MD   44, // The manipulated value is always limited by an upper and lower limit.
  PV_FAC       := 1.000000e+000, // = Standard, scaling Proces value between 0...100% = 0...27648
  PV_OFF       := 0.000000e+000, // no Offset
  LMN_FAC      := 1.000000e+000, // = Standard, scaling Output value between 0...100% = 0...27648
  LMN_OFF      := 0.000000e+000, // no Offset
  LMN          := MD   48, // The effective manipulated value is output in floating point format at the "manipulated value" output.
  LMN_PER      := PQW  512, // The manipulated value in the I/O format is connected to the controller at the "manipulated value peripheral" output.
  QLMN_HLM     := M    10.4,
  QLMN_LLM     := M    10.5,
  LMN_P        := MD   52,
  LMN_I        := MD   56,
  LMN_D        := MD   60,
  PV           := MD   64,
  ER           := MD   68);
  BE
;
END_ORGANIZATION_BLOCK

ORGANIZATION_BLOCK OB 100
TITLE = "Complete Restart"
VERSION : 0.1

VAR_TEMP
OB100_EV_CLASS : BYTE ; //16#13, Event class 1, Entering event state, Event logged in diagnostic buffer
OB100_STRTUP   : BYTE ; //16#81/82/83/84 Method of startup
OB100_PRIORITY : BYTE ; //Priority of OB Execution
OB100_OB_NUMBR : BYTE ; //100 (Organization block 100, OB100)

```

```
OB100_RESERVED_1 : BYTE ; //Reserved for system
OB100_RESERVED_2 : BYTE ; //Reserved for system
OB100_STOP : WORD ; //Event that caused CPU to stop (16#4xxx)
OB100_STRT_INFO : DWORD ; //Information on how system started
OB100_DATE_TIME : DATE_AND_TIME ; //Date and time OB100 started
```

```
END_VAR
```

```
BEGIN
```

```
NETWORK
```

```
TITLE =Place first values to the controller
```

```
OPN DB 41;
SET ;
= M 10.0;
= M 10.1;
```

```
L 1.000000e+001;
T MD 12;
L 2.000000e+000;
T MD 24;
L T#20S;
T MD 28;
L T#10S;
T MD 32;
L 1.000000e+002;
T MD 40;
BE ;
```

```
END_ORGANIZATION_BLOCK
```