Project Documentation

File: CtrlX_Module_03_2025.project

Date: 24.02.2025

Profile: ctrlX PLC 1.20.7

Table des matières

1 DUT: E_OperationBaseDL

```
{attribute 'qualified_only' := ''}
{attribute 'strict' := ''}
TYPE E_OperationBaseDL :
(
    Idle := 99,
    Init := 1,
    InOpLevelLow := 2,
    InOpLevelOk := 3,
    InOpLevelHigh := 4,
    Error
);
END_TYPE

```

2 POU: FB_O300_DL

```
FUNCTION_BLOCK FB_O300_DL
VAR_INPUT
    Enable : BOOL;
    HighThreshold : REAL;
    LowThreshold : REAL;
    DefaultOut : REAL;
END_VAR
VAR_IN_OUT
    hw : UA_O300_DL;
END_VAR
VAR_OUTPUT
    InOperation : BOOL;
    Value : REAL;
    HighLimit : BOOL;
    LowLimit : BOOL;
    Error : BOOL;
    ErrorId : WORD;
END_VAR
VAR
    eOperationBaseDL : E_OperationBaseDL;
    inputConverted : REAL;
END_VAR

```

```
// Read Input
inputConverted := hw.Value * 0.1;

CASE eOperationBaseDL OF
    E_OperationBaseDL.Idle              :
        IF Enable THEN
            eOperationBaseDL := E_OperationBaseDL.Init;
        END_IF
    E_OperationBaseDL.Init              :
        IF NOT Enable THEN
            eOperationBaseDL := E_OperationBaseDL.Idle;
```

```
12              ELSIF hw.A OR hw.Q THEN
13                  eOperationBaseDL := E_OperationBaseDL.Error;
14              ELSIF inputConverted > HighThreshold THEN
15                  eOperationBaseDL := E_OperationBaseDL.InOpLevelHigh;
16              ELSIF inputConverted < LowThreshold THEN
17                  eOperationBaseDL := E_OperationBaseDL.InOpLevelLow;
18              ELSE
19                  eOperationBaseDL := E_OperationBaseDL.InOpLevelOk;
20              END_IF
21          E_OperationBaseDL.InOpLevelLow :
22              IF NOT Enable THEN
23                  eOperationBaseDL := E_OperationBaseDL.Idle;
24              ELSIF hw.A OR hw.Q THEN
25                  eOperationBaseDL := E_OperationBaseDL.Error;
26              ELSIF inputConverted >= LowThreshold THEN
27                  eOperationBaseDL := E_OperationBaseDL.InOpLevelOk;
28              END_IF
29          E_OperationBaseDL.InOpLevelOk  :
30              IF NOT Enable THEN
31                  eOperationBaseDL := E_OperationBaseDL.Idle;
32              ELSIF hw.A OR hw.Q THEN
33                  eOperationBaseDL := E_OperationBaseDL.Error;
34              ELSIF inputConverted > HighThreshold THEN
35                  eOperationBaseDL := E_OperationBaseDL.InOpLevelHigh;
36              ELSIF inputConverted < LowThreshold THEN
37                  eOperationBaseDL := E_OperationBaseDL.InOpLevelLow;
38              END_IF
39          E_OperationBaseDL.InOpLevelHigh   :
40              IF NOT Enable THEN
41                  eOperationBaseDL := E_OperationBaseDL.Idle;
42              ELSIF hw.A OR hw.Q THEN
43                  eOperationBaseDL := E_OperationBaseDL.Error;
44              ELSIF inputConverted <= HighThreshold THEN
45                  eOperationBaseDL := E_OperationBaseDL.InOpLevelOk;
46              END_IF
47          E_OperationBaseDL.Error            :
48              IF NOT Enable THEN
49                  eOperationBaseDL := E_OperationBaseDL.Idle;
50              END_IF
51      END_CASE
52
53      InOperation := (eOperationBaseDL = E_OperationBaseDL.InOpLevelHigh) OR
54                     (eOperationBaseDL = E_OperationBaseDL.InOpLevelLow)  OR
55                     (eOperationBaseDL = E_OperationBaseDL.InOpLevelOk);
56
57      HighLimit := (eOperationBaseDL = E_OperationBaseDL.InOpLevelHigh);
58      LowLimit  := (eOperationBaseDL = E_OperationBaseDL.InOpLevelLow);

59
60      Error := (eOperationBaseDL = E_OperationBaseDL.Error);
61
62      IF (eOperationBaseDL = E_OperationBaseDL.InOpLevelHigh) OR
63         (eOperationBaseDL = E_OperationBaseDL.InOpLevelLow)  OR
64         (eOperationBaseDL = E_OperationBaseDL.InOpLevelOk)   THEN
65          ErrorId := 16#0;
66      ELSIF (eOperationBaseDL = E_OperationBaseDL.Error) AND
```

```
67            hw . A                                                THEN
68          ErrorId  :=  16#2 ;
69      ELSIF  ( eOperationBaseDL  =  E_OperationBaseDL . Error )  AND
70            hw . Q                                                THEN
71          ErrorId  :=  16#1 ;
72      ELSE
73          ErrorId  :=  16#3 ;
74      END_IF
75
76      IF  InOperation  THEN
77          Value  :=  inputConverted ;
78      ELSE
79          Value  :=  DefaultOut ;
80      END_IF
81
82
83
```

3 POU: PLC_PRG

```
1       ///
2       ///         www.hevs.ch
3       ///         Institut Systemes Industriels
4       ///         Project:        Projet No: PW_03
5       ///         Author:               Cedric Lenoir
6       ///         Date:               2024 January 23
7       ///
8       ///         Summary:        Main program for Practical Work 03.
9       PROGRAM  PLC_PRG
10      VAR
11          diMyLoop :  DINT ;
12          testMode :  BOOL ;
13          fbO300_DL :  FB_O300_DL ;
14      END_VAR
15
```

```
1       diMyLoop  :=  diMyLoop  +  1 ;
2
3       fbO300_DL ( hw  :=  GVL_Abox . uaAboxInterface . uaO300_DL_Optic ) ;
4
5
```