

## Project Documentation

File: Finale 12072020.project

Date: 13.07.2020

Profile: CODESYS V3.5 SP16

## Inhaltsverzeichnis

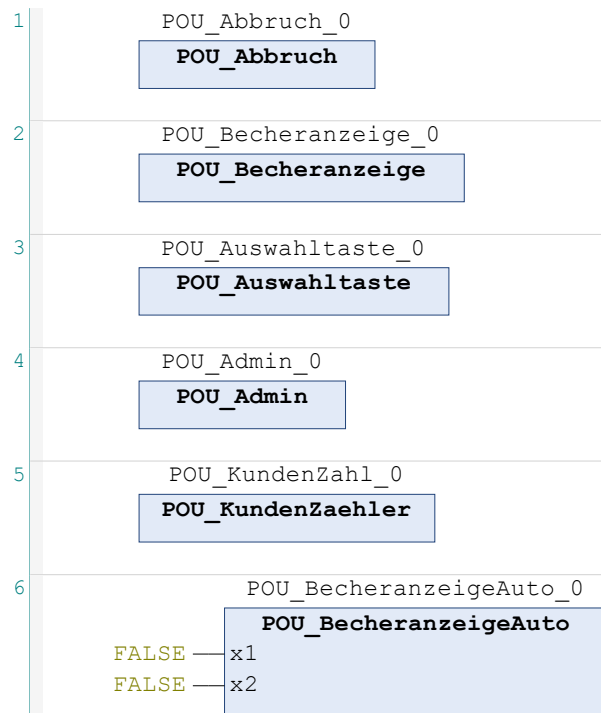
1	Globale Variablenliste: Globale_Variablen	3
2	POU: PLC_PRG	3
3	POU: POU_Abbruch	5
4	POU: POU_Admin	6
5	POU: POU_Auswahltaste	7
6	POU: POU_Becheranzeige	9
7	POU: POU_BecheranzeigeAuto	10
8	POU: POU_Kaffee_oder_Tee	13
9	POU: POU_KundenZaehler	14

# 1 Globale Variablenliste: Globale\_Variablen

```
1      {attribute 'qualified_only'}
2      VAR_GLOBAL
3          // Variable für die Simulation der Becheranzeige
4          Fuellstand : INT := 0 ;           //Simulationsvariable Füllstand
5          Leer : BOOL ;                     //Simulationsvariable Leer
6          Full : BOOL ;                     //Simulationsvariable Voll
7          // Eingänge
8          S1 : BOOL ;                       // Kaffee
9          S2 : BOOL ;                       // Tee
10         S3 : BOOL ;                       // Zucker
11         S4 : BOOL ;                       // Milch
12         S5 : BOOL ;                       // Münzeinwurf
13         S6 : BOOL ;                       // Admin
14         S9 : BOOL ;                       // Abbruch
15         // Anzeigen
16         H1 : BOOL ;                       // Kaffee
17         H2 : BOOL ;                       // Tee
18         H3 : INT := 0 ;                   // Zucker
19         H4 : INT := 0 ;                   // Milch
20         H5 : BOOL ;                       // Münzeinwurf
21         H6 : BOOL ;                       // Admin
22         H9 : BOOL ;                       // Abbruch
23         // Variable für die "Kundenzähler"
24         KundenZahl : WORD ;               //Zählvariable für Getränkeähler
25         disp_Kundenzahl : STRING ;        //Textvariable für Getränkeähler
26         // Displayvariable
27         Display : STRING ;                //Textvariable für Display
28         //Verriegelungsvariablen für Milch und Zucker
29         Flanke1 : BOOL := TRUE ;          //Flankenerkennung
30         Flanke2 : BOOL := TRUE ;          //Flankenerkennung
31         G1 : BOOL := TRUE ;               //Verriegelungsvariable
32         G2 : BOOL := TRUE ;               //Verriegelungsvariable
33         // Füllsensor
34         HB2 : BOOL ;                      // Signalisierung Becher entnehmen
35         B2 : BOOL ;                       // Taster für Becherentnahme
36         B1 : BOOL ;                       //Signalisierung Becher voll
37         //
38         x2 : BOOL ;                       // wird für TON benutzt
39         K1 : BOOL := TRUE ;               // Verriegelungsvariable
40
41         // Variable für die Dosierung Milch und Zucker
42         Anzahl1 : INT ;                   //Zutatenzähler Zucker
43         Anzahl2 : INT ;                   //Zutatenzähler Milch
44     END_VAR
45
```

## 2 POU: PLC\_PRG

```
1  PROGRAM PLC_PRG
2  VAR
3
4
5      // abgelaufene Zeit
6      Zeit : TIME ;
7      Zeitint : INT := 0 ;
8
9      POU_Becheranzeige : POU_Becheranzeige ;
10     POU_0 : POU_Kaffee_oder_Tee ;
11     POU_Abbruch_0 : POU_Abbruch ;
12     POU_Becheranzeige_0 : POU_Becheranzeige ;
13     POU_Auswahltaste_0 : POU_Auswahltaste ;
14     POU_Admin_0 : POU_Admin ;
15     POU_KundenZahl_0 : POU_KundenZaehler ;
16     POU_BecheranzeigeAuto_0 : POU_BecheranzeigeAuto ;
17
18
19
20 END_VAR
21
```



7

POU\_0

**POU\_Kaffee\_oder\_Tee**

### 3 POU: POU\_Abbruch

```
1  FUNCTION_BLOCK POU_Abbruch
2  VAR_INPUT
3  END_VAR
4  VAR_OUTPUT
5  END_VAR
6  VAR
7  END_VAR
8
9
10 //hier werden alle Variablen bei Abbruch in den Ausgangszustand versetzt
11
```

---

```
1  Globale_Variablen . H9 := Globale_Variablen . S9 ;
2  IF ( Globale_Variablen . H9 = TRUE AND Globale_Variablen . Leer = FALSE
3  AND Globale_Variablen . B1 = FALSE ) THEN
4
5      Globale_Variablen . K1 := TRUE ;
6      Globale_Variablen . S1 := FALSE ;
7      Globale_Variablen . S2 := FALSE ;
8      Globale_Variablen . H1 := FALSE ;
9      Globale_Variablen . H2 := FALSE ;
10     Globale_Variablen . H3 := 0 ;
11     Globale_Variablen . H4 := 0 ;
12     Globale_Variablen . H5 := FALSE ;
13     Globale_Variablen . S5 := FALSE ;
14     Globale_Variablen . Display := 'Abbruch' ;
15     Globale_Variablen . Leer := FALSE ;
16     Globale_Variablen . Full := FALSE ;
17     Globale_Variablen . G1 := TRUE ;
18
19 END_IF
20 // In der oberen If Bedingung wird gefragt ob die Abbruchtaster S9 betätigt
21 // wurde , wenn ja
22 // dann soll alles abgebrochen werden und die Lampe H9 leuchtet so lange S9
23 // betätigt ist .
24
25 IF Globale_Variablen . H9 = FALSE THEN
26
27     Globale_Variablen . Display := 'Bereit' ;
28 END_IF
```

## 4 POU: POU\_Admin

```
1  FUNCTION_BLOCK POU_Admin
2  VAR_INPUT
3  END_VAR
4  VAR_OUTPUT
5  END_VAR
6  VAR
7
8  Merker3 : BOOL ;
9  Merker4 : BOOL ;
10
11 END_VAR
12
```

---

```
1  Merker3 := Globale_Variablen . H5 ;
2
3  // die lokale Variable Merker3 dient dazu zu prüfen ob der Taster S5
  // betätigt ist und schaltet entsprechend H5 (Leuchte)
4  // wenn ja , dann soll das Programm in die 2. Bedingung springen ,
5  //wenn nein , dann muss die erste If Bedingung ausgeführt werden
6
7  IF ( Globale_Variablen . S6 = TRUE AND Merker3 = FALSE ) THEN
8      Globale_Variablen . H6 := TRUE ;
9      Globale_Variablen . H5 := FALSE ;
10     Globale_Variablen . Display := 'Admin' ;
11
12 END_IF
13
14
15 //in der oberen If Bedingung wird geprüft welcher Taster zuerst betätigt
  //wird,
16 //genauer gesagt wenn die Lampe H6 true ist dann wird die Lampe H5 auf False
  //gesetzt und damit blockiert.
17
18 Merker4 := Globale_Variablen . H6 ;
19
20 IF ( Globale_Variablen . S5 = TRUE AND Merker4 = FALSE ) THEN
21     Globale_Variablen . H5 := TRUE ;
22     Globale_Variablen . H6 := FALSE ;
23
24 END_IF
25 //in der oberen If Bedingung wird geprüft welcher Taster zuerst betätigt
  //wird,
26 //genauer gesagt wenn die Lampe H5 true ist dann wird die Lampe H6 auf False
  //gesetzt und damit blockiert.
27 Globale_Variablen . H6 := Globale_Variablen . S6 ;
28
```

## 5 POU: POU\_Auswahltaste

```
1  FUNCTION_BLOCK POU_Auswahltaste
2  VAR_INPUT
3  END_VAR
4  VAR_OUTPUT
5  END_VAR
6  VAR
7  Merker1 : BOOL ;
8  Anzahl3 : INT ;
9  Merker2 : BOOL ;
10 Anzahl4 : INT ;
11
12
13 END_VAR
14
```

---

```
1  // H3 Zuckeranzeige
2
3  Globale_Variablen . Flanke1 := Globale_Variablen . S3 AND NOT Merker1 ;
4  Merker1 := Globale_Variablen . S3 ;
5
6  IF Globale_Variablen . Flanke1 = 1 AND Globale_Variablen . G1 = TRUE THEN
7      Anzahl3 := Anzahl3 + 1 ;
8      Globale_Variablen . H3 := Anzahl3 ;
9  END_IF
10
11
12 IF Merker1 = TRUE THEN
13
14 IF Anzahl3 <= 3 THEN
15     CASE Anzahl3 OF
16
17         1 : Globale_Variablen . Display := '1x Zucker' ;
18         2 : Globale_Variablen . Display := '2x Zucker' ;
19     ELSE
20         Globale_Variablen . H3 := 0 ;
21
22     END_CASE
23
24
25 END_IF
26 END_IF
27
28 IF ( Anzahl3 >= 3 OR Globale_Variablen . S9 ) THEN
29     Anzahl3 := 0 ;
30 END_IF
31
32 // H4 Milchanzeige
33 Globale_Variablen . Flanke2 := Globale_Variablen . S4 AND NOT Merker2 AND
Globale_Variablen . G1 ;
34 Merker2 := Globale_Variablen . S4 ;
35
```



```

36
37   IF Globale_Variablen . Flanke2 = 1 AND Globale_Variablen . G2 = TRUE THEN
38       Anzahl14 := Anzahl14 + 1 ;
39       Globale_Variablen . H4 := Anzahl14 ;
40   END_IF
41
42
43   IF Merker2 = TRUE THEN
44
45       IF Anzahl14 <= 3 THEN
46           CASE Anzahl14 OF
47
48               1 : Globale_Variablen . Display := '1x Milch' ;
49               2 : Globale_Variablen . Display := '2x Milch' ;
50           ELSE
51               Globale_Variablen . H4 := 0 ;
52
53           END_CASE
54
55       END_IF
56   END_IF
57
58   IF ( Anzahl14 >= 3 OR Globale_Variablen . S9 ) THEN
59       Anzahl14 := 0 ;
60   END_IF
61
62
63   IF ( Anzahl13 >= 3 OR Globale_Variablen . S9 ) THEN
64       Anzahl13 := 0 ;
65   END_IF
66
67

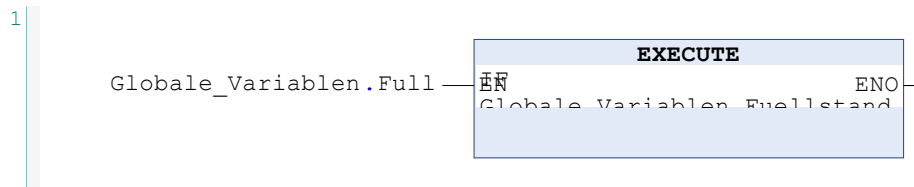
```

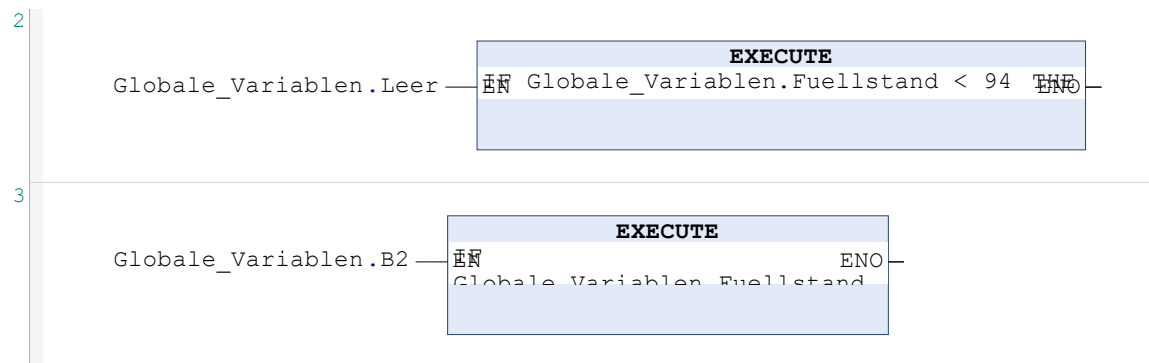
## 6 POU: POU\_Becheranzeige

```

1   FUNCTION_BLOCK POU_Becheranzeige
2
3   VAR
4
5   END_VAR
6
7   // Dieses Modul dient der Simulation der Füllanzeige
8

```





## 7 POU: POU\_BecheranzeigeAuto

```
1  FUNCTION_BLOCK POU_BecheranzeigeAuto
2  VAR_INPUT
3      x1 : BOOL ;
4      x2 : BOOL ;
5      TON : TON ; //Einschaltverzögerungsglied
6
7  END_VAR
8  VAR_OUTPUT
9  END_VAR
10 VAR
11 END_VAR
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
```

---

```
1  IF Globale_Variablen . H5 AND ( Globale_Variablen . H1 OR Globale_Variablen .
H2 ) THEN
2      x1 := TRUE ;
3      Globale_Variablen . K1 := FALSE ;
4      END_IF
5
6  IF Globale_Variablen . Full = TRUE THEN
7      Globale_Variablen . Full := FALSE ;
8
9  END_IF
10 // In der oberen If Bedingung wird gefragt ob der Becher am Anfang der
Simulation voll ist ,
11 //wenn ja wird er "geleert" das bedeutet, der Becher wird entnommen .
12
13 IF Globale_Variablen . Leer = TRUE THEN
14     Globale_Variablen . Leer := FALSE ;
15     Globale_Variablen . B2 := FALSE ;
16
17 END_IF
18
19 TON ( IN := x1 , PT := T#3S , Q=> Globale_Variablen . x2 , ET=> ) ;
20
21 IF Globale_Variablen . x2 = TRUE THEN
22     Globale_Variablen . Leer := TRUE ;
23     END_IF
24 // "Globale_Variablen.Leer " ist zuständig für das Füllen des Bechers und
25 // ist dann True wenn im normalen Modus (Müneinwurf) oder im Adminmodus eine
der Getränke gewählt wird
26
27 IF Globale_Variablen . Fuellstand = 80 THEN
28     Globale_Variablen . S1 := FALSE ;
29     Globale_Variablen . S2 := FALSE ;
30
31     Globale_Variablen . H1 := FALSE ;
32     Globale_Variablen . H2 := FALSE ;
33     Globale_Variablen . K1 := TRUE ;
34     Globale_Variablen . B1 := TRUE ;
35     Globale_Variablen . Leer := FALSE ;
```

```
36         Globale_Variablen . Display := 'Becher entnehmen' ;
37     END_IF
38     // In der Oberen If Bedingung wird der maximale Füllstand des Bechers
    geprüft und wenn dieser erreicht wird
39     // darf der Becher nicht mehr gefüllt werden
40     IF Globale_Variablen . K1 = FALSE THEN
41         Globale_Variablen . HB2 := FALSE ;
42         Globale_Variablen . B2 := FALSE ;
43     END_IF
44
45     IF Globale_Variablen . HB2 = TRUE THEN
46         Globale_Variablen . Full := True ;
47         Globale_Variablen . H1 := FALSE ;
48         Globale_Variablen . H2 := FALSE ;
49         Globale_Variablen . H3 := 0 ;
50         Globale_Variablen . H4 := 0 ;
51         Globale_Variablen . S3 := FALSE ;
52         Globale_Variablen . S4 := FALSE ;
53         Globale_Variablen . H5 := FALSE ;
54         Globale_Variablen . S5 := FALSE ;
55         Globale_Variablen . Display := 'Bereit' ;
56         Globale_Variablen . B1 := FALSE ;
57     END_IF
58     Globale_Variablen . HB2 := Globale_Variablen . B2 ;
59     // In unserem Programm simuliert B2 die "Becherentnahme". Wenn der Becher
    entnommen wurde ,
60     // müssen alle vorher gewählten Variablen auf den Anfangszustand
    zurückgesetzt werden .
61
62
63     // ADMIN MODUS BECHERANZEIGE
64
65     IF Globale_Variablen . H6 AND ( Globale_Variablen . S1 OR Globale_Variablen
    . S2 ) THEN
66         Globale_Variablen . Leer := True ;
67     END_IF
68     // "Globale_Variablen.Leer " ist zuständig für das Füllen des Bechers und
69     // ist dann True wenn im Normalmodus (Müneinwurf) oder Adminmodus eines der
    Getränke gewählt wurde
70
71     IF Globale_Variablen . Full = TRUE THEN
72         Globale_Variablen . Full := FALSE ;
73
74     END_IF
75     // In der oberen If Bedingung wird gefragt ob der Becher am Anfang der
    Simulation voll ist ,
76     //wenn ja wird er "geleert" das Bedeutet der Becher wird entnommen .
77
78
79     IF Globale_Variablen . Fuellstand = 80 THEN
80         Globale_Variablen . Leer := FALSE ;
81     END_IF
82     // In der Oberen If Bedingung wird der maximale Füllstand des Bechers
    geprüft und wenn dieser erreicht ist
83     // darf der Becher nicht mehr gefüllt werden
84
```

```
85     IF Globale_Variablen . HB2 = TRUE THEN
86         Globale_Variablen . Full := True ;
87         Globale_Variablen . H1 := FALSE ;
88         Globale_Variablen . H2 := FALSE ;
89         Globale_Variablen . Anzahl1 := 0 ;
90         Globale_Variablen . Anzahl2 := 0 ;
91         Globale_Variablen . H3 := 0 ;
92         Globale_Variablen . H4 := 0 ;
93         Globale_Variablen . S3 := FALSE ;
94         Globale_Variablen . S4 := FALSE ;
95         Globale_Variablen . H5 := FALSE ;
96         Globale_Variablen . S5 := FALSE ;
97         Globale_Variablen . H3 := 0 ;
98         Globale_Variablen . H4 := 0 ;
99         Globale_Variablen . B1 := FALSE ;
100        Globale_Variablen . Fuellstand := 0 ;
101
102        Globale_Variablen . G1 := TRUE ;
103
104        END_IF
105
106
107
108
109     IF Globale_Variablen . H5 = TRUE THEN
110         Globale_Variablen . S3 := FALSE ;
111         Globale_Variablen . S6 := FALSE ;
112         Globale_Variablen . S4 := FALSE ;
113         Globale_Variablen . Flankel := FALSE ;
114         Globale_Variablen . G1 := FALSE ;
115
116     END_IF
117
118
119     // In unserem Programm simuliert B2 die "Becherentnahme" , wenn der Becher
120     // entnommen wurde ,
121     // müssen alle vorher gewählten Variablen auf den Anfangszustand
122     // zurückgesetzt werden .
```

## 8 POU: POU\_Kaffee\_oder\_Tee

```
1     FUNCTION_BLOCK POU_Kaffee_oder_Tee
2     VAR_INPUT
3
4         Merker1 : BOOL ;
5         Merker2 : BOOL ;
6
7     END_VAR
8     VAR_OUTPUT
9
10    END_VAR
11    VAR
12
13    END_VAR
```

```
1      Merker2 := Globale_Variablen . H2 ;
2      // die Lokale Variable Merker2 dient dazu , ob der Taster S2 betätigt ist
      // und dementsprechend leuchtet H2
3      // wenn ja , dann soll das Programm in die 2. If Bedingung springen ,
4      //wenn nein , dann wird die erste If Bedingung ausgeführt
5
6      IF ( Globale_Variablen . S1 = TRUE AND Merker2 = FALSE AND
      Globale_Variablen . H5 ) THEN
7          Globale_Variablen . H1 := TRUE ;
8          Globale_Variablen . S2 := FALSE ;
9      END_IF
10
11      //in der oberen If Bedingung wird geprüft welcher Taster zuerst betätigt
      //wird,
12      //genauer gesagt wenn der taster S1 true ist dann wird der Taster S2 auf
      //False gesetzt und damit blockiert bzw. umgekehrt
13
14
15      IF Globale_Variablen . H1 = TRUE THEN
16          Globale_Variablen . Display := 'Kaffee kommt' ;
17      END_IF
18
19
20      Merker1 := Globale_Variablen . H1 ;
21      IF ( Globale_Variablen . S2 AND Merker1 = FALSE and Globale_Variablen . H5
      ) THEN
22          Globale_Variablen . H2 := TRUE ;
23          Globale_Variablen . S1 := FALSE ;
24
25      END_IF
26      IF Globale_Variablen . H2 = TRUE THEN
27          Globale_Variablen . Display := 'Tee kommt' ;
28      END_IF
29
30
31
32
33
34
35
36
37
38
39
40
41
```

## 9 POU: POU\_KundenZaehler

```
1  FUNCTION_BLOCK POU_KundenZaehler
2  VAR_INPUT
3      CTU : CTU ;           // Counter Variable für Aufwärtszähler
4      CV : INT ;           // Zählervariable für Kundenzahl
5  END_VAR
6  VAR_OUTPUT
7  END_VAR
8  VAR
9  END_VAR
10
```

---

```
1
2      CTU (
3      CU := ( Globale_Variablen . B1 AND Globale_Variablen . H5 ) ,
4      RESET := FALSE ,
5      PV := 1000 ,
6      Q => ,
7      CV => Globale_Variablen . Kundenzahl ) ;
8      Globale_Variablen . disp_Kundenzahl := ANY_TO_STRING ( Globale_Variablen .
Kundenzahl ) ;
9
```