

Assignment 2:

Problem solving with methodologies using Beckhoff twincat3

Course: ASE: 9416 (2016)

Course name: Discrete Automation System



Submitted By:

MD. SHAHEDUL ALAM (267943)

PRASUN BISWAS (267948)

USAMAH (267590)

Objective:

Our objective is to use different methodologies to represent the problem and then solve it in PLC programs using Beckhoff TwinCAT.

Description:

As per the requirement we developed an Algorithm and wrote our program with Ladder Diagram.

We followed some simple steps for completing project successfully. They are sequentially

discussed: -

- 1) First, we developed an algorithm as per requirement of our task.
- 2) From the algorithm, we developed the Flow chart of the Algorithm and then made a rough sketch of PLC logic.
- 3) Then we opened a new project file in TwinCAT 3 and defined our required variables and their types, like BOOL, INT etc.
- 4) After the completion of Structure text coding, we clicked the build option and then activated our project and clicked the "Run" option. We checked output by changing its original values from true to false and vice versa and checked whether there is any error or not.

Change of States:

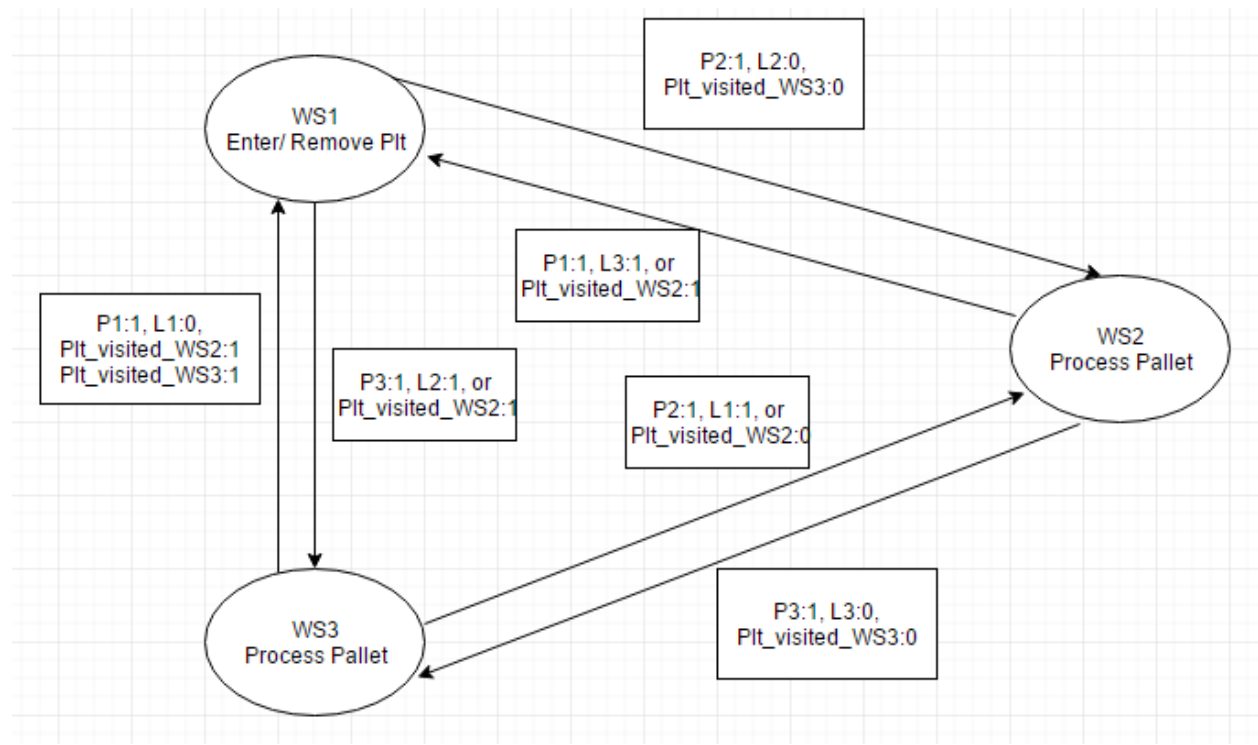


Diagram: State change and condition

Algorithm for the solution:

1. Start the System
2. Check if the Workstation is empty to enter the Pallet with the sensor Lx.
3. Check if the Position sensor detects a Pallet in front of the Work Station:
4. Check if the Pallet is already have been in workstation for processing
5. If all the condition of above is satisfied as per requirements the enter in the work station.
6. If condition not satisfied as per requirements then proceed to the next Work station.

***** TYPE Declaration*****

TYPE sPallet :

STRUCT

 bVisited_WS1:BOOL;

 bVisited_WS2:BOOL;

 bVisited_WS3:BOOL;

 bPresent_in_WS1:BOOL;

 bPresent_in_WS2:BOOL;

 bPresent_in_WS3:BOOL;

 bPlt_remove:BOOL;

END_STRUCT

END_TYPE

***** Variables Declaration*****

PROGRAM MAIN

VAR_INPUT

L1:BOOL;

L2:BOOL;

L3:BOOL;

P1:BOOL;

P2:BOOL;

P3:BOOL;

END_VAR

VAR_OUTPUT

S1:BOOL;

S2: BOOL;

S3:BOOL;

BA1:BOOL;

BA2:BOOL;

BA3:BOOL;

// to show message about the process in Work Station

sMessage1:STRING;

sMessage2:STRING;

sMessage3:STRING;

END_VAR

VAR CONSTANT

iPallet_count: UINT:=4;

END_VAR

VAR

fbTimer1:TON;

fbTimer2:TON;

fbTimer3:TON;

fbTimer_plt_on_conv1:TON;

fbTimer_plt_on_conv2:TON;

fbTimer_plt_on_conv3:TON;

i:UINT;

sPallet: ARRAY[0..iPallet_count] OF sPallet;

END_VAR

*****main program *****

// condition for Workstation 2

IF P2 AND (NOT L2) THEN

 FOR i:=0 TO iPallet_count BY 1 DO

 IF (NOT sPallet[i].bVisited_WS2) AND (NOT sPallet[i].bPresent_in_WS1) AND
(NOT sPallet[i].bPresent_in_WS3) THEN

 S2:=TRUE;

 L2:=TRUE;

 sPallet[i].bPresent_in_WS2:=TRUE;

 sMessage2:='Pallet in WS3 and being Processed';

 IF fbTimer2.Q THEN

 sMessage2:='Processing Finished in WS3';

 sPallet[i].bVisited_WS2:=TRUE;

 END_IF

 EXIT;

 END_IF

 END_FOR

END_IF

IF L2 THEN

 BA2:= FALSE;

 ELSIF NOT L2 THEN

 BA2:=TRUE;

END_IF

IF S2 THEN

fbTimer2(IN:=S2,PT:=T#5S);

ELSIF NOT S2 THEN

fbTimer_plt_on_conv2(IN:=NOT S2,PT:=T#3S);

END_IF

IF fbTimer_plt_on_conv2.Q THEN

sMessage2:='Moving to the next Work station';

END_IF

IF fbTimer2.Q THEN

 S2:=FALSE;

 L2:=FALSE;

END_IF

```

// condition for Workstation 3
IF P3 AND (NOT L3) THEN
    FOR i:=0 TO iPallet_count BY 1 DO
        IF (NOT sPallet[i].bVisited_WS3) AND (NOT sPallet[i].bPresent_in_WS1) AND
(NOT sPallet[i].bPresent_in_WS2) THEN
            S3:=TRUE;
            L3:=TRUE;
            sPallet[i].bPresent_in_WS3:=TRUE;
            sMessage3:='Pallet in WS3 and being Processed';

            IF fbTimer3.Q THEN
                sMessage3:='Processing Finished in WS3';
                sPallet[i].bVisited_WS3:=TRUE;
            END_IF
            EXIT;
        END_IF
    END_FOR
END_IF
IF L3 THEN
    BA3:= FALSE;
    ELIF NOT L3 THEN
        BA3:=TRUE;
    END_IF
IF S3 THEN
    fbTimer3(IN:=S3,PT:=T#5S);
    ELIF NOT S3 THEN
        fbTimer_plt_on_conv3(IN:=NOT S3,PT:=T#3S);
    END_IF

    IF fbTimer_plt_on_conv3.Q THEN
        sMessage3:='Moving to the next Work station';
    END_IF

    IF fbTimer3.Q THEN
        S3:=FALSE;
        L3:=FALSE;
    END_IF

```

```

// condition for Workstation 1
IF P1 AND (NOT L1) THEN
    FOR i:=0 TO iPallet_count BY 1 DO
        IF (sPallet[i].bVisited_WS3) AND (sPallet[i].bVisited_WS2) AND (NOT
sPallet[i].bPresent_in_WS2) AND (NOT sPallet[i].bPresent_in_WS3) THEN
            S1:=TRUE;
            L1:=TRUE;
            sPallet[i].bPresent_in_WS1:=TRUE;
            sMessage1:='Pallet in WS1 and being Processed';

            IF fbTimer1.Q THEN
                sPallet[i].bVisited_WS1:=TRUE;
                sPallet[i].bPlt_remove:=TRUE; // Pallet removed from the system
            END_IF
            EXIT;
        END_IF
    END_FOR
END_IF
IF L1 THEN
    BA1:= FALSE;
    ELSIF NOT L1 THEN
        BA1:=TRUE;
    END_IF
IF S1 THEN
    fbTimer1(IN:=S1,PT:=T#5S);
    ELSIF NOT S1 THEN
        fbTimer_plt_on_conv1(IN:=NOT S1,PT:=T#3S);
    END_IF

    IF fbTimer_plt_on_conv1.Q THEN
        sMessage1:='Moving to the next Work station';
    END_IF

    IF fbTimer1.Q THEN
        sMessage1:='Removed From WS1';
        S1:=FALSE;
        L1:=FALSE;
    END_IF
    IF i> 4 THENs
        i:=0;
    END_IF

```


Difficulties faced and few things to mention:

1. Firstly, as we had some experience with Ladder Logic gained through in studies of the courses and outside of curriculum Practical work and as we had very little knowledge and skill with Structure Text coding, we tried to solve the problem with combination of Ladder logic and Structural Text coding. After many trial and error we had to ditch Ladder Logic completely.
2. We managed to write a code where all the conditions, mentioned in the stated problem, are checked. But still have some Problem with the implementation.
3. As the MOTOR, will be always ON while the system running we skipped the Motor on variable considering systems on equivalent to motor running.
4. Doing this assignment was a hard journey, but was extremely necessary one , because now I am confident that doing any other project with Structure Text will be easy.

Work Load Distribution:

Our group was formed with 3 members. Everyone's effort was noteworthy in almost every aspect. But as we had to work in Lab there was shortage of Computer with TwinCAT3 so we had to shuffle the responsibilities in between us. Names are written as descending priorities. That means person who has done the most work in that field his name is written first.

- Algorithm: Prasun Biswas, Md. Shahedul Alam, Usamah
- Structure Text Code: Prasun Biswas, Usamah, Md. Shahedul Alam,
- Report Writing: Prasun Biswas, Usamah, Shahedul Alam
- TwinCAT3 Research: Prasun Biswas, Usamah, Shahedul Alam
- Flowchart: Md. Shahedul Alam, Prasun Biswas, Usamah
- State Chart: Shahedul Alam, Prasun Biswas, Usamah