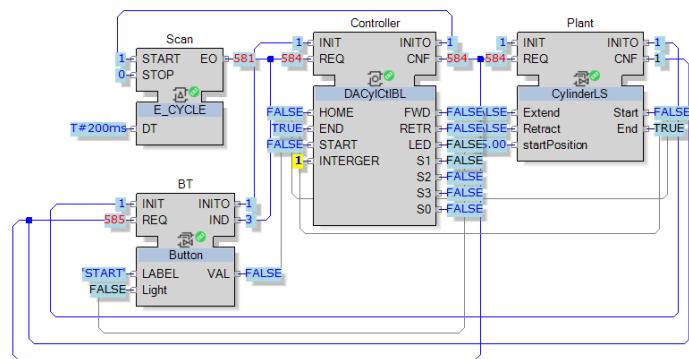


INIT BOOLEAN Sx IFTHENELSE Newstate

```

1  ALGORITHM BOOLEAN IN ST:
2  (* Add your comment (as per IEC 61131-3) here
3
4  *)
5  S2:=S2 AND NOT END OR S1 AND START;
6  S3:=S3 AND NOT HOME OR S2 AND END;
7  S1:=S1 AND NOT START OR S3 AND HOME;
8
9 // Outputs
10 LED := S1;
11 FWD := S2;
12 RETR := S3;
13 END_ALGORITHM

```



- Boolean functions-based implementation done. The pictures demonstrate the problem well: Cylinder doesn't retract as wanted and all the states are false.

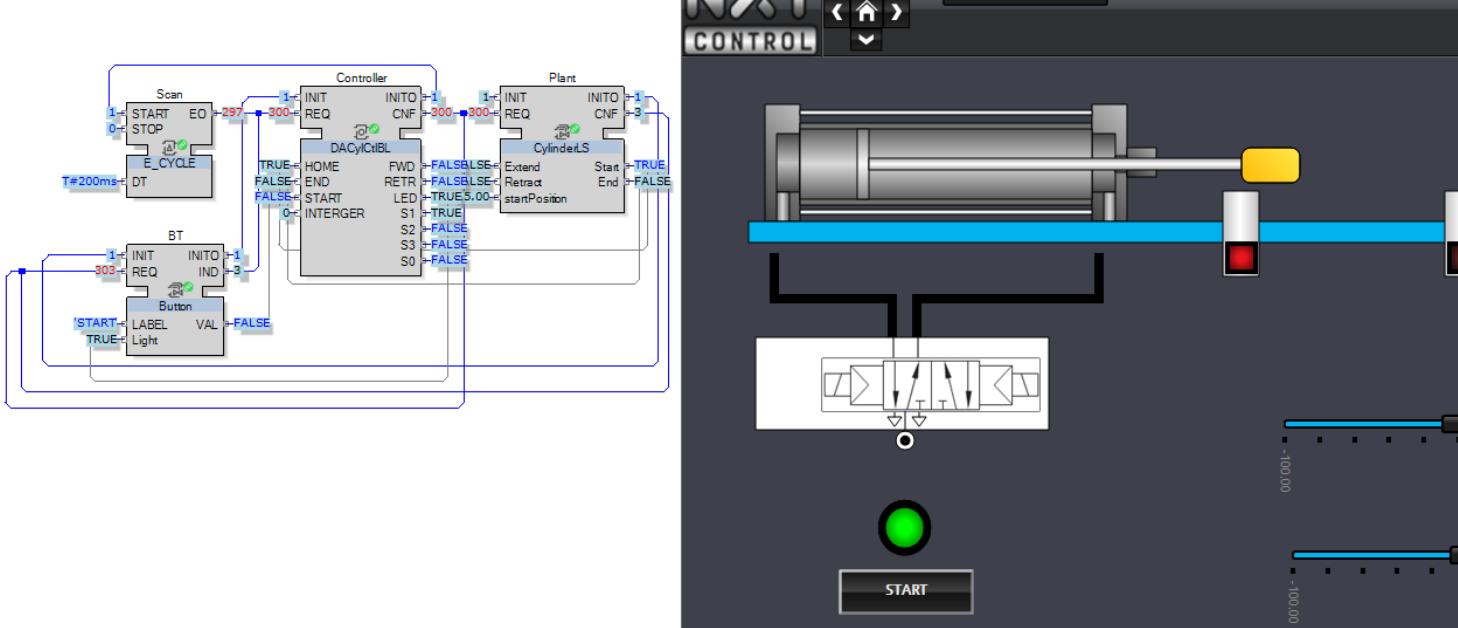
2.

```

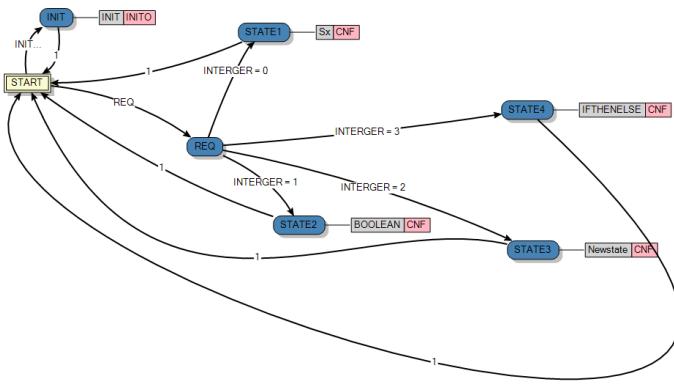
INIT BOOLEAN Sx IFTHENELSE Newstate
1  ALGORITHM Sx IN ST:
2  (* Add your comment (as per IEC 61131-3) here
3
4  *)
5  S1x := 1;
6  S2x := 0;
7  S3x := 0;
8  S2x := S2 AND NOT END OR S1 AND START;
9  S3x := S3 AND NOT HOME OR S2 AND END;
10 S1x := S1 AND NOT START OR S3 AND HOME;
11 // Outputs
12 LED := S1x;
13 FWD := S2x;
14 RETR := S3x;
15 S1 := S1x;
16 S2 := S2x;
17 S3 := S3x;
18
19 END_ALGORITHM

```

Name	Type	Attr...	Initial v...	With	Attr	Comment
EventInputs						
INIT						
REQ						
<new interface>						
EventOutputs						
INITO						
CNF						
<new interface>						
InputVars						
HOME	BOOL					
END	BOOL					
START	BOOL					
INTERGER	INT					
<new variable>						
OutputVars						
FWD	BOOL					
RETR	BOOL					
LED	BOOL					
S1	BOOL					
S2	BOOL					
S3	BOOL					
S0	BOOL					
<new variable>						
InternalVars						
State	INT					
S1x	BOOL					
S2x	BOOL					
S3x	BOOL					
S0x	BOOL					
<new variable>						



Here we have added S_{1x}, S_{2x} and S_{3x} to fix the issue. We have also added a new integer input to allow switching between different algorithms. The cylinder now functions as wanted.



INIT	BOOLEAN	Sx	IFTHENELSE	Newstate
1	ALGORITHM	Newstate IN ST:		
2		(* Add your comment (as per IEC 61131-3) here		
3		*)		
4		;		
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				
16				
17				
18				
19				END_ALGORITHM

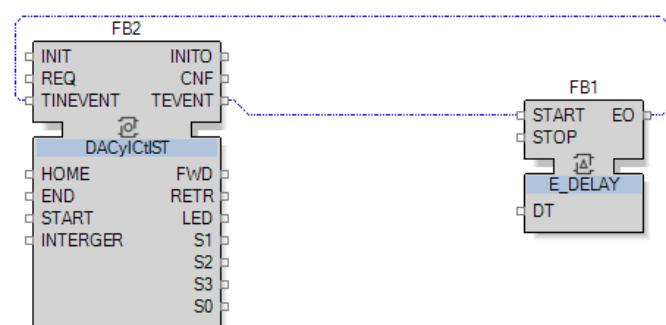
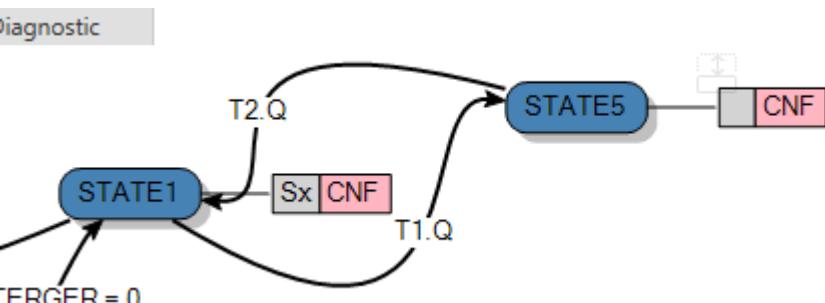
```

S2x := S2 AND NOT END OR S1 AND START;
S3x := S3 AND NOT HOME OR S2 AND END;
S1x := S1 AND NOT START OR S3 AND HOME OR S0 AND HOME;
S0x := S0 AND NOT HOME;

LED := S1x;
FWD := S2x;
RETR := S3x;
S0 := S0x;
S1 := S1x;
S2 := S2x;
S3 := S3x;

```

For the timer we can create a sister state:



In the interface I created new EVENINPUT and EVENTOUTPUT because E_Delay needs these.

EventInputs			
INIT	HOME, END, S...	<none>	
REQ	HOME, END, S...	<none>	
TINEVENT		<none>	
<new interface>			
EventOutputs			
INITO	FWD, RETR, LE...	<none>	
CNF	FWD, RETR, LE...	<none>	
TEVENT	S1	<none>	