PZZ-AC [FB16]

PZZ-AC Properties							
General							
Name	PZZ-AC	**Number**	16	**Type**	FB	**Language**	SCL
Numbering	Automatic						
Information							
Title		**Author**		**Comment**		**Family**	
Version	0.1	**User-defined ID**					

Name	Data type	Default value	Retain	Accessible from HMI/OPC UA/Web API	Writable from HMI/OPC UA/Web API	Visible in HMI engineering	Setpoint	Supervision	Comment
▼ Input									
start	String	''	Non-retain	True	True	True	False		
▼ Output									
R1	Bool	false	Non-retain	True	True	True	False		
W	Bool	false	Non-retain	True	True	True	False		
R2	Bool	false	Non-retain	True	True	True	False		
DUTYCYKL	Time	T#0ms	Non-retain	True	True	True	False		
DUTYCYKL2	Time	T#0ms	Non-retain	True	True	True	False		
InOut									
▼ Static									
STAV	Int	3	Non-retain	True	True	True	False		
PWM_IN	Bool	false	Non-retain	True	True	True	False		
PWM_OUT	Bool	false	Non-retain	True	True	True	False		
PWM1_IN	Bool	false	Non-retain	True	True	True	False		
PWM1_OUT	Bool	false	Non-retain	True	True	True	False		
timer_state	Int	0	Non-retain	True	True	True	False		
PWMs_on	Bool	false	Non-retain	True	True	True	False		
▼ T_W	TON_TIME		Non-retain	True	True	True	False		
PT	Time	T#0ms	Non-retain	True	True	True	False		
ET	Time	T#0ms	Non-retain	True	False	True	False		
IN	Bool	false	Non-retain	True	True	True	False		
Q	Bool	false	Non-retain	True	False	True	False		
▼ T_R	TON_TIME		Non-retain	True	True	True	True		
PT	Time	T#0ms	Non-retain	True	True	True	False		
ET	Time	T#0ms	Non-retain	True	False	True	False		
IN	Bool	false	Non-retain	True	True	True	False		
Q	Bool	false	Non-retain	True	False	True	False		
Temp									
Constant									

```
0001 IF #start = 'OPEN' THEN
0002   #STAV := 2;
0003   #PWMs_on := 1;
0004   #timer_state := 0;
0005 ELSIF #start = 'STOP' THEN
0006   #STAV := 1;
0007   #PWMs_on := 1;
0008   #timer_state := 0;
0009 ELSIF #start = 'OFF' THEN
0010   #STAV := 3;
0011   #PWMs_on := 0;
0012 END_IF;
0013
0014 #DUTYCYKL := T#750ms;
0015 #DUTYCYKL2 := T#750ms;
0016
0017 #DUTYCYKL := T#375ms;
0018 #DUTYCYKL2 := T#375ms;
0019
0020 CASE #STAV OF
0021
0022   1:
0023     #W := false;
0024
0025     #R1 := NOT #R1;
0026     #R2 := NOT #R2;
0027
0028     #T_R(IN := TRUE,
0029        PT := T#375ms);
0030     IF #T_R.Q THEN
0031       #T_R.IN:=FALSE; // Transition to idle state after 3 seconds
0032     END_IF;
0033
0034   2:
0035     #R1 := false;
0036     #R2 := false;
```

```
0037
0038      #W := NOT #W;
0039
0040      #T_W(IN := TRUE,
0041         PT := T#750ms);
0042      IF #T_W.Q THEN
0043        #T_W.IN:=FALSE; // Transition to idle state after 3 seconds
0044      END_IF;
0045
0046    3:
0047      #PWM_IN := FALSE;
0048      #PWM1_IN := false;
0049      #R1 := false;
0050      #R2 := false;
0051      #W := FALSE;
0052
0053  END_CASE;
0054
0055  CASE #timer_state OF
0056    0:
0057      #PWM_IN := TRUE;
0058      IF #PWM_OUT = 1 THEN
0059        #PWM_IN := FALSE;
0060        #timer_state := 1;
0061      END_IF;
0062
0063    1:
0064      #PWM1_IN := TRUE;
0065      IF #PWM1_OUT = 1 THEN
0066        #PWM1_IN := FALSE;
0067        #timer_state := 0;
0068      END_IF;
0069
0070  END_CASE;
```

Symbol	Address	Type	Comment
#DUTYCYKL		Time	
#DUTYCYKL2		Time	
#PWM_IN		Bool	
#PWM_OUT		Bool	
#PWM1_IN		Bool	
#PWM1_OUT		Bool	
#PWMs_on		Bool	
#R1		Bool	
#R2		Bool	
#start		String	
#STAV		Int	
#T_R		IEC_Timer	
#T_R.IN		Bool	
#T_R.Q		Bool	
#T_W		IEC_Timer	
#T_W.IN		Bool	
#T_W.Q		Bool	
#timer_state		Int	
#W		Bool	