

'4x2 Taster mit ATTiny2313A von Günther Wille (c)2020

'Ta - Ebene 1 (Schalter 10 - links)

,

'Für CARSON STICK 14 Kanal # 201606 (Kanäle 11 - 14 / Taster)

'Am PORTD.0 Signal von Kanal 10 (3 St. Schalter)

'Am Eingang der PORTD.1 - D.4 des Tiny2313A wird das PWM Signal (Kanal 11 - 14) vom Fernsteuerempfänger eingelesen und anschließend auf 8 Ports entsprechend ausgegeben.

'Das eingelesene Signal hat bei meiner Fernsteuerung BYTE-Werte zw. >> 39 und 72 <<.

'Über eine Abfrageschleife (10 x IF THEN) werden die Ausgangsports gesetzt.

'Die Ports werden nur bei Tastung auf "1" gesetzt, dann nach 5 mS wieder auf "0".

'Statt der LED's können später FET's diverse Verbraucher / Sounds EIN Tasten.

\$regfile = "attiny2313.dat"

\$crystal = 1000000

Led1 Alias Portb.7	'LED1
Led2 Alias Portb.6	'LED2
Led3 Alias Portb.5	'LED3
Led4 Alias Portb.4	'LED4
Led5 Alias Portb.3	'LED5
Led6 Alias Portb.2	'LED6
Led7 Alias Portb.1	'LED7
Led8 Alias Portb.0	'LED8
Led9 Alias Portd.6	'LED9 Ebene 2 Tb
Led10 Alias Portd.5	'LED10 Ebene 3 Tc

Config Pind.0 = Input	'Input PWM 1
Config Pind.1 = Input	'Input PWM 2
Config Pind.2 = Input	'Input PWM 3
Config Pind.3 = Input	'Input PWM 4
Config Pind.4 = Input	'Input PWM 5

Config Portb.7 = Output	'Ausgang Port 1
Config Portb.6 = Output	'Ausgang Port 2
Config Portb.5 = Output	'Ausgang Port 3
Config Portb.4 = Output	'Ausgang Port 4
Config Portb.3 = Output	'Ausgang Port 5
Config Portb.2 = Output	'Ausgang Port 6
Config Portb.1 = Output	'Ausgang Port 7
Config Portb.0 = Output	'Ausgang Port 8
Config Portd.6 = Output	'Ausgang Port 9
Config Portd.5 = Output	'Ausgang Port 10

Puls1 Alias Pind.0	'Abfrage Schalter 10 (Ebene 1, 2, 3)
Pulsport1 Alias Portd.0	
Puls2 Alias Pind.1	'Abfrage Taster 11
Pulsport2 Alias Portd.1	
Puls3 Alias Pind.2	'Abfrage Taster 12
Pulsport3 Alias Portd.2	
Puls4 Alias Pind.3	'Abfrage Taster 13
Pulsport4 Alias Portd.3	
Puls5 Alias Pind.4	'Abfrage Taster 14
Pulsport5 Alias Portd.4	

Dim Messen1 As Byte	'Variable für Eingangswert
Dim Messen2 As Byte	
Dim Messen3 As Byte	
Dim Messen4 As Byte	
Dim Messen5 As Byte	

Led1 = 0

Led2 = 0

Led3 = 0
Led4 = 0
Led5 = 0
Led6 = 0
Led7 = 0
Led8 = 0
Led9 = 0
Led10 = 0

Do 'Beginn der Endlos-Schleife

***** Pulslänge einlesen *****

Pulsein Messen1 , Pind , 0 , 1
Pulsein Messen2 , Pind , 1 , 1
Pulsein Messen3 , Pind , 2 , 1
Pulsein Messen4 , Pind , 3 , 1
Pulsein Messen5 , Pind , 4 , 1

If Messen1 > 80 Or Messen1 < 30 Then 'Rücksetzen, wenn Wert unplausibel
'oder kein Empfangssignales
 Led9 = 0
 Led10 = 0
End If

If Messen1 > 46 And Messen1 < 60 Then 'Schalter 10 (mitte)
 Led9 = 1 'EIN
 Waitms 5
Else
 Led9 = 0
End If

If Messen1 > 65 Then 'Schalter 10 (rechts)
 Led10 = 1 'EIN
 Waitms 5
Else
 Led10 = 0
End If

If Messen1 > 30 And Messen1 < 45 Then 'Schalter 10 (links)

 If Messen2 > 30 And Messen2 < 45 Then 'Taster 11a
 Led1 = 1 'EIN
 Waitms 5
 Else
 Led1 = 0 'AUS
 End If

 If Messen2 > 65 Then 'Taster 11b
 Led2 = 1 'EIN
 Waitms 5
 Else
 Led2 = 0
 End If

 If Messen3 > 30 And Messen3 < 45 Then 'Taster 12a
 Led3 = 1 'EIN
 Waitms 5
 Else
 Led3 = 0
 End If

 If Messen3 > 65 Then 'Taster 12b
 Led4 = 1 'EIN
 Waitms 5
 Else
 Led4 = 0

End If

If Messen4 > 30 And Messen4 < 45 Then 'Taster 13a
 Led5 = 1 'EIN
 Waitms 5

Else
 Led5 = 0

End If

If Messen4 > 65 Then 'Taster 13b
 Led6 = 1 'EIN
 Waitms 5

Else
 Led6 = 0

End If

If Messen5 > 30 And Messen5 < 45 Then 'Taster 14a
 Led7 = 1 'EIN
 Waitms 5

Else
 Led7 = 0

End If

If Messen5 > 65 Then 'Taster 14b
 Led8 = 1 'EIN
 Waitms 5

Else
 Led8 = 0

End If

End If

Loop

'Programmende Ende