

Zelf PIC's programmeren

Overzicht van enkele veel gebruikte commando's

Variabele

- bit 0 of 1, low of high
- byte 0 t/m 255
- sbyte -128 t/m 127
- word 0 t/m 65.535
- sword -32.768 t/m 32.767
- dword 0 t/m 4.294.967.296
- sdword -2.147.483.648 t/m 2.147.483.647

Het declareren van variabelen moet plaatsvinden voordat ze gebruikt worden.

Voorbeeld

```
var byte lamp3 -- lamp3 is gedeclareerd als een byte
```

Constanten

- const byte demo = 5

Het voordeel van constanten is, dat deze minder geheugen gebruiken.

Random

De random mogelijkheden

- Random_Byte Een willekeurig getal tussen 0 en 255
- Random_Word Een willekeurig getal tussen 0 en 65535
- Dice Een willekeurig getal tussen 1 en 6 (dobbelsteen)

Voorbeeld

```
var byte waarde3  
waarde3= Random_Byte
```

Input

```
pin_A0_Direction= input  
var volatile bit Uit is pin_A2  
alias PWW is pin_A2  
mA=ADC_read(0) pin A0
```

pin_A0 wordt nu als input gezet
pin_A2 krijgt een andere naam
A2 pin_A2 krijgt een andere naam, PWW
wordt analoog ingelezen tussen 0 en 1023

Output

```
Pin_A2_Direction= output
```

For

```
For 8 loop  
x=x+2  
end loop  
Deze loop wordt zo 8 keer herhaald
```

While

While $x < 23$ loop

$x = x + 1$

end loop

Deze loop wordt net zolang herhaald totdat x meer is dan 22.

If then else

If $x == 3$ then $y = 2$ else $y = 6$ end if

Block

Block

$x = 6$

$y = 7$

end block

Repeat Until

Repeat

$x = x + 1$

Until $x == 7$

Herhaal totdat je aan de voorwaarde voldoet (hier moet x gelijk zijn aan 7).

Deze loop wordt minimaal een keer doorlopen.

Case

Case 3 of

1: $x = 5$

2: $x = 7$

3: block $x = 9$ end block

4: $x = 11$

otherwise $x = 2$

End case

x krijgt in dit geval de waarde 9, omdat er gekozen is voor optie 3, en daar staat dat x naar de waarde 9 gaat. De 3 hier mag ook een variabele zijn bv. een byte.

Delay (er zijn meer delays)

delay_10us(n) n moet een byte zijn

delay_10us(n) n moet een byte zijn

delay_1ms(n) n moet een word zijn

delay_100ms(n) n moet een word zijn

delay_1s(n) n moet een word zijn

Voorbeeld

delay_10us(6) wacht 60 micro-seconde

delay_1s(18) wacht 18 seconde

Procedure

Procedure wisselA is

Var byte a

a=a+1

End procedure

De waarde a is alleen binnen de procedure in gebruik.

Procedure wisselB (byte in a) is

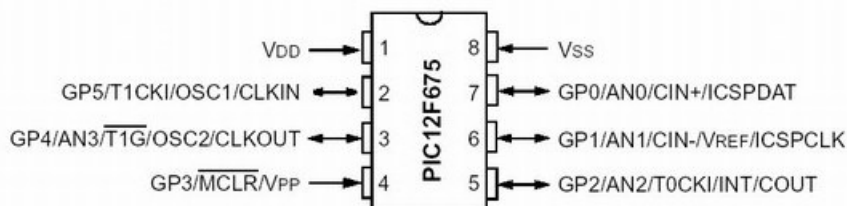
a=a+1

End procedure

Lege formats

Bij het starten van een nieuw project kan het makkelijk zijn om te starten met een lege format.

Lege format voor de PIC 12F675



```
Include 12f675
```

```
-- target PICmicro
```

```
pragma target clock 4_000_000
pragma target OSC INTOSC_NOCLKOUT
pragma target WDT disabled
pragma target MCLR internal
pragma target BROWNOUT disabled
enable_digital_io()
```

```
-- oscillator frequency
-- Internal oscillator
-- no watchdog
-- make MCLR pin available I/O
-- no brownout
-- disable analog I/O (if any)
```

```
include delay
include random
```

```
-- het laden van de delay's
-- het laden van de random's
```

```
var byte waardeX
```

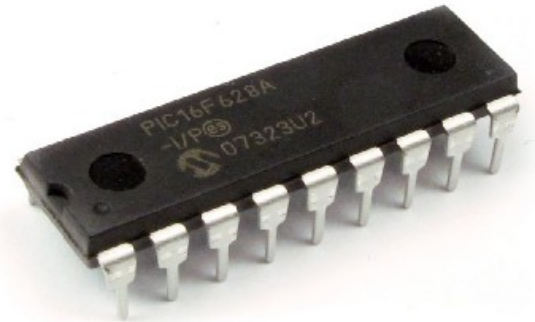
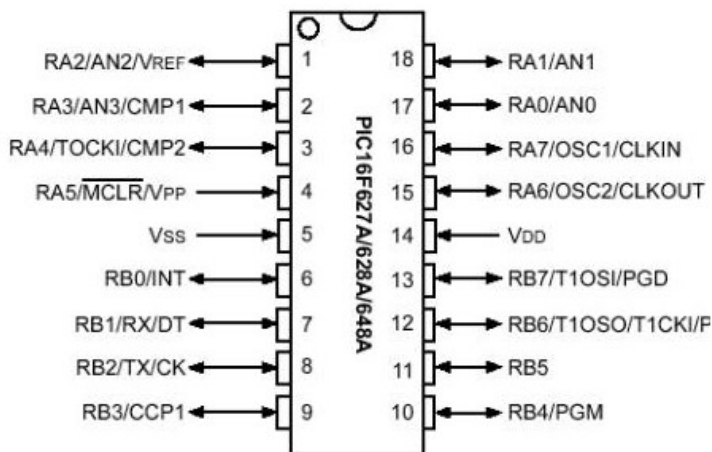
```
pin_A0_direction = Output    alias LED1 is pin_A0
pin_A1_direction = Input     alias Knop1 is pin_A1
```

```
Forever loop
```

```
-- maak hier je programma
-- maak hier je programma
-- maak hier je programma
```

```
end loop
```

Lege format voor de PIC 16F628a



```
include 16f628a
```

```
pragma target clock 20_000_000  
pragma target OSC HS  
pragma target WDT DISABLED  
pragma target BROWNOUT DISABLED  
pragma target LVP DISABLED  
pragma target MCLR EXTERNAL  
pragma target fuses 0x3D02  
const Serial_HW_Baudrate = 19200  
include serial_hardware
```

```
enable_digital_io()
```

```
Include Delay
```

```
pin_A4_direction = INPUT  
pin_B0_direction = OUTPUT
```

```
alias INgang4 is pin_A4  
alias led0 is pin_B0
```

```
var byte WaardeX
```

```
forever loop  
    -- maak hier je programma  
    -- maak hier je programma  
    -- maak hier je programma  
end loop
```

Opm: De PIC16F628a maakt gebruik van een uitwendig kristal van 20.000kHz.

```
-- oscillator frequency  
-- crystal or resonator  
-- watchdog  
-- no brownout reset  
-- no low voltage programming  
-- external reset
```