

DHT20 of AHT20 I2C Temperatuur en Vochtigheid sensor

' Voorbeeld van interface met de sensor DHT20 zonder een bibliotheek te gebruiken, maar alleen bestaande I2C commando's / functies met MMBasic.
' <http://www.aosong.com/userfiles/files/media/Data%20Sheet%20DHT20.pdf>
' Deze subroutine gevonden op het Annex Basic forum en aangepast voor MMBasic.

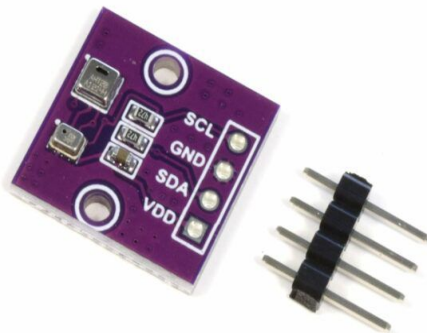
```
OPTION SYSTEM I2C GP4,GP5          ' I2C bus SDA=GP4, SCL=GP5
OPTION RTC AUTO ENABLE

te = 0
hu = 0
Do
  dht20_init                        ' Start de sensor
  dht20_read te,hu                  ' Lees temperatuur en vochtigheid.
  Print "Vocht:";Str$(hu,2,1);" %", "Temp:";Str$(te,2,2);Chr$(176);"C",
  Print "Tijd:";Left$(Time$,5)
  Pause 60000                       ' Wacht 60 Sec.
Loop
End

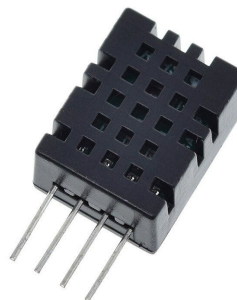
Sub dht20_init
  I2C Write &h38,0,1,&hbe          ' Begin (resetten).
  Pause 100                         ' Pauze na reset 100mS.
End Sub

Sub dht20_read(temp,hum)
  Local Integer arr(6)              ' Array gebruikt voor het lezen.
  I2C Write &h38,0,3,&hac,&h33,&h00 ' Activeer de meting.
  Pause 80                          ' Wachtijd gevraagd voor het lezen 80mS.
  I2C READ &h38,0,6, arr()          ' Lees de gegevens.
  If (arr(0) And 128) Then           ' Arr(0) is de status.
    Print "Gegevens niet beschikbaar, verhoog de wachttijd"
  Else
    hum = ((arr(1) << 12) + (arr(2) << 4) + (arr(3) >> 4)) / 10485.76
    temp = (((arr(3) And &h0f) << 16) + (arr(4) << 8) + arr(5)) / 1048576
    temp = temp * 200 - 50
  End If
End Sub

> run
Vocht:59.7 %    Temp:17.74°C    Tijd:15:41
```



AHT20 + BMP280 sensor



DHT20 sensor

```

' Option SYSTEM SPI GP18,GP19,GP16
' Option SYSTEM I2C GP4,GP5          ' GP4=SDA , GP5=SCL
' Option LCDPANEL SSD1306I2C, LANDSCAPE
' Option SDCARD GP17
' Option RTC AUTO ENABLE
' RTC SETTIME 2023,3,26,21,28,0 jaar,maand,dag,uur,min,sec RTC tijd instellen.
' Zonder RTC tijd: Time$="hh:mm:ss" en Date$="dd-mm-jjjj" Bij power on opnieuw.
' Code dht20 komt van AnnexBasic en is geschikt gemaakt voor MMBasic.
te = 0
hu = 0
' CLS                                ' Wis Oled scherm.
' Backlight 20                       ' Backlight 0 - 100.
Do
    dht20_init                        ' Start de sensor.
    dht20_read te,hu                 ' Lees temperatuur en vochtigheid.
    Print "DHT20: ", "Vocht:";Str$(hu,2,1);" %","Temp:";Str$(te,2,1);Chr$(176);
    Print "C","Tijd:";Left$(Time$,5);" ";dagw$(Date$);" "; Date$
    ' Text 0 , 0, dagw$(Date$)        ' Dagw$(Date$) is dag van de week.
    ' Text 20, 0, Left$(Date$,5)
    ' Text 80, 0, Left$(Time$,5)
    ' Text 0 ,20, "Temp.:      C"
    ' Text 81,20, Chr$(96)            ' Let op Chr$(96) is graden teken.
    ' Text 50,20, Str$(te,2,1)        ' Str$(te,2,1) is temperatuur xx.x C
    ' Text 0 ,40, "Vocht:      %"
    ' Text 50,40, Str$(hu,2,1)        ' Str$(hu,2,1) is vochtigheid xx.x %
    Pause 60000                      ' Wacht 20 Sec.
Loop
End
Sub dht20_init                        ' DHT20 sub routine.
    I2C Write &h38,0,1,&hbe          ' Begin (resetten).
    Pause 100                        ' Pauze na reset 100mS.
End Sub
Sub dht20_read(temp,hum)
    Local integer arr(6)              ' Array gebruikt voor het lezen.
    I2C Write &h38,0,3,&hac,&h33,&h00 ' Activeer de meting.
    Pause 80                          ' Wachtijd gevraagd voor het lezen 80mS.
    I2C READ &h38,0,6, arr()          ' Lees de gegevens.
    If (arr(0) And 128) Then           ' Arr(0) is de status.
        Print "Gegevens niet beschikbaar, verhoog de wachttijd"
    Else
        hum = ((arr(1) << 12) + (arr(2) << 4) + (arr(3) >> 4)) / 10485.76
        temp = (((arr(3) And &h0f) << 16) + (arr(4) << 8) + arr(5)) / 1048576
        temp = temp * 200 - 50
    EndIf
End Sub
Function dagw$(dt$)                  ' Functie dag van de week bepalen NL Ma-Zo
    Local INTEGER d,m,y,jd,weekday
    d = Val(Mid$(dt$,1,2))            ' dd-xx-xxxx.
    m = Val(Mid$(dt$,4,2))            ' xx-mm-xxxx.
    y = Val(Mid$(dt$,7,4))            ' xx-xx-jjjj.
    If m < 3 Then m = m + 12 : y = y - 1
    jd = d + Fix(365.25 * (y+4716)) + Fix(30.6001 * (m + 1)) - y\100 + y\400 + 4
    weekday = (jd Mod 7) + 1
    dagw$ = Mid$(" Ma Di Wo Do Vr Za Zo ", weekday * 3, 3)
End Function

```

LCDI2CLib test voor bijna elk type PCF8574 I2C naar LCD-module

PicoMite MMBasic Version 5.07.07b5

OPTION SYSTEM SPI GP18,GP19,GP16

OPTION SYSTEM I2C GP4,GP5, SLOW

OPTION COLOURCODE ON

OPTION DISPLAY 40, 90

OPTION LCDPANEL SSD1306I2C, LANDSCAPE ' 128x64 Oled scherm

OPTION SDCARD GP22

OPTION RTC AUTO ENABLE

' Deze subroutine gevonden op het MMBasic forum en geschikt gemaakt.

' Gebruikt PCF8574 - 8 bit I2C Port Expander en zonder 4K7 pullups aan 5V

' I2CAddress is &H20=20x4 LCD, &H38=16x2 LCD of zie: I2C_Scanner.bas

' PCF8574-poort is als volgt bedraad:

' P0 - RS

' P1 - RW

' P2 - EN

' P3 - Achtergrondverlichting aan/uit

' P4 - D4 LCD

' P5 - D5 LCD

' P6 - D6 LCD

' P7 - D7 LCD-scherm

' LCDI2C_Init &H38,1,0,2,3 ' I2CAddr, D0~D3=0 D4~D7=1, RS=0, EN=2, BL=3.

' LCDI2C_BackLightOn ' Zet de LCD-achtergrondverlichting AAN.

' LCDI2C_BackLightOff ' Zet de LCD-achtergrondverlichting UIT.

' LCDI2C(LineNum, CharPos, Text\$) ' Line, Char, Text\$=Tekens of Commando.

' LCDI2C_CLEAR() ' Wist het LCD.

' LCDI2C_CMD(Byte) ' Stuur een COMMAND byte naar de LCD, dwz RS=0

' LCDI2C_DATA(Byte) ' Stuur een DATA byte naar de LCD, dwz RS=1

' LCDI2C_DirectSend(Byte) ' Stuur een BYTE naar het LCD.

' * Intern gebruikt door de lib *

' LCDI2C_WireWrite(Byte) ' Schrijf een byte op de I2C-bus naar de LCD.

' WAITFORAKEY(MSG\$) ' Wacht op een toets input data.

' LCDI2C_Init &H38,1,0,2,3 ' I2CAddr, D0~D3=0 D4~D7=1, RS=0, EN=2, BL=3

LCDI2C_Init &H3F, 1, 0, 2, 3 ' I2CAddr=&H20=20x4, &H38=16x2, &3F=16x2.

' Hoofd programma.

Do ' Met een 4x20 2004 LCD scherm.

LCDI2C_BackLightOn ' Achtergrond LED aan.

LCDI2C_CMD(64) ' Aangepast teken Chr\$(0) is CMD Reg. 64-71.

LCDI2C_DATA(7): LCDI2C_DATA(5): LCDI2C_DATA(7): LCDI2C_DATA(0) ' Byte 1-4.

LCDI2C_DATA(0): LCDI2C_DATA(0): LCDI2C_DATA(0): LCDI2C_DATA(0) ' Byte 5-8.

LCDI2C 1, 1, "BuitenT:R 18.3 C" ' LCDI2C Line, Pos, Data\$

LCDI2C 1,15, Chr\$(0) ' Zet graden teken op Line=1, Pos=15, Chr\$(0)

LCDI2C 2, 1, "BinnenT: 20.6 C" ' LCDI2C Line, Pos, Data\$

LCDI2C 2,15, Chr\$(0) ' Zet graden teken op Line=2, Pos=15, Chr\$(0)

Pause 10000 ' Wacht 10 seconde.

Loop

End

```

' **** Alle subroutines voor lcd 16x2 en 20x4 ****
' Wacht op een toets input data.
Sub WaitForAKey(Msg$)
    If Msg$ = "" Then Msg$ = "Press any key to continue"
    Print Msg$;
    Do
        Loop While Inkey$ = ""
    Print
End Sub
' Line=1-4, Char=1-16, Text$=CHR$ + ALLE CHARACTERS
Sub LCDI2C(LineNum, CharPos, Text$)
    Local I
    If LineNum = 1 Then I = (&H80 + CharPos - 1)
    If LineNum = 2 Then I = (&HC0 + CharPos - 1)
    If LineNum = 3 Then I = (&H94 + CharPos - 1)
    If LineNum = 4 Then I = (&HD4 + CharPos - 1)
    LCDI2C_CMD(I)
    For I = 1 To Len(Text$)
        LCDI2C_DATA(Asc(Mid$(Text$, I, 1)))
    Next I
End Sub
' Wis het LCD scherm
Sub LCDI2C_Clear()
    LCDI2C_CMD(&H01)
End Sub
'
Sub LCDI2C_Init I2CAAddr, NibPos, RSBitNo, ENBitNo, BLBitNo
    Dim LCDI2C_BackLight
    LCDI2C_BackLight = 2 ^ BLBitNo
    Dim LCDI2C_LCDBackLight
    LCDI2C_LCDBackLight = LCDI2C_BackLight
    Dim LCDI2C_I2CAAddr
    LCDI2C_I2CAAddr = I2CAAddr
    Dim LCDI2C_RSDataMask
    LCDI2C_RSDataMask = 2 ^ RSBitNo
    Dim LCDI2C_EMask
    LCDI2C_EMask = 2 ^ ENBitNo
    Dim LCDI2C_NibPos
    LCDI2C_NibPos = NibPos
    LCDI2C_CMD(&H33)
    LCDI2C_CMD(&H32)
    LCDI2C_CMD(&H28)
    LCDI2C_CMD(&H0C)
    LCDI2C_CMD(&H06)
    LCDI2C_CMD(&H01)
    LCDI2C_BackLightOff()
End Sub
'
Sub LCDI2C_Close()
    I2C CLOSE
End Sub

```

```

Sub LCDI2C_BacklightOn
    LCDI2C_LCDBacklight = LCDI2C_Backlight
    LCDI2C_WireWrite(0)
End Sub

Sub LCDI2C_BacklightOff
    LCDI2C_LCDBacklight = &H00
    LCDI2C_WireWrite(0)
End Sub

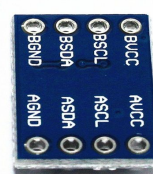
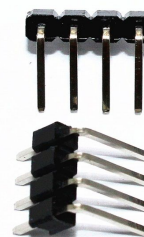
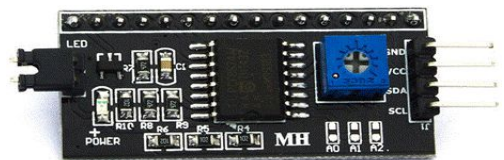
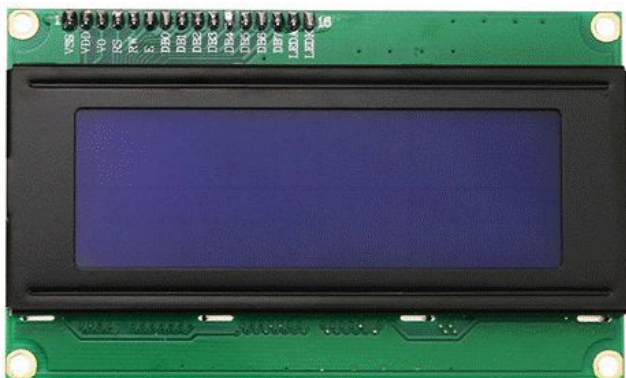
Sub LCDI2C_CMD(Byte)
    LCDI2C_DirectSend(Byte And &HF0)
    LCDI2C_DirectSend((Byte And &H0F) * 16)
End Sub

Sub LCDI2C_DATA(Byte)
    LCDI2C_DirectSend((Byte And &HF0) Or LCDI2C_RSDataMask)
    LCDI2C_DirectSend(((Byte And &H0F) * 16) Or LCDI2C_RSDataMask)
End Sub

Sub LCDI2C_DirectSend(Byte)
    Local B, B1
    If LCDI2C_NibPos = 0 Then
        B1 = Byte \ 16
    Else
        B1 = Byte
    EndIf
    B = B1 Or LCDI2C_EMask Or LCDI2C_LCDBacklight
    I2C WRITE LCDI2C_I2CAAddr, 0, 1, B
    I2C WRITE LCDI2C_I2CAAddr, 0, 1, B1 Or LCDI2C_LCDBacklight
End Sub

Sub LCDI2C_WireWrite(Byte)
    I2C WRITE LCDI2C_I2CAAddr, 0, 1, Byte Or LCDI2C_LCDBacklight
End Sub

```



Aangepaste grafische afbeeldingen maken op 1602/2004 LCD

Dit werkt op HD44780 KS0066 SPLC780-controllerchips met 1,2 of 4 regels tekst.

De karakters zijn gemaakt op een 5X8 raster.

Hier is een aangepast karakter van een bel.

	16	8	4	2	1	
1		*				= 4
2		*	*	*		= 14(8+4+2=14)
3		*	*	*		= 14
4		*	*	*		= 14
5	*	*	*	*	*	= 31(16+8+4+2+1=31)
6						= 0
7		*				= 4
8						= 0

Er zijn 8 aangepaste tekens:

CHR\$(0)	heeft het commando LCD CMD	64	' Byte 1= 64, 2= 65, ... 8= 71
CHR\$(1)	heeft het commando LCD CMD	72	' Byte 1= 72, 2= 73, ... 8= 79
CHR\$(2)	heeft het commando LCD CMD	80	' Byte 1= 80, 2= 81, ... 8= 87
CHR\$(3)	heeft het commando LCD CMD	88	' Byte 1= 88, 2= 89, ... 8= 95
CHR\$(4)	heeft het commando LCD CMD	96	' Byte 1= 96, 2= 97, ... 8=103
CHR\$(5)	heeft het commando LCD CMD	104	' Byte 1=104, 2=105, ... 8=111
CHR\$(6)	heeft het commando LCD CMD	112	' Byte 1=112, 2=113, ... 8=119
CHR\$(7)	heeft het commando LCD CMD	120	' Byte 1=120, 2=121, ... 8=127

Dus om het Bell-teken hierboven af te drukken, gebruikt u

```
LCD CMD 64 : LCD DATA 4, 14, 14, 14, 31, 0, 4, 0
LCD 1, 1, CHR$(0)
```

Om een ander aangepast teken af te drukken, gebruikt u de volgende CHR\$ -

```
LCD CMD 72 : LCD DATA X,X,X,X,X,X,X,X
LCD 1, 2, CHR$(1)
```

Denk eraan als u de PicoMite gebruikt om BITBANG LCD te gebruiken.

De moeite waard om toe te voegen, je hoeft het slechts één keer te doen :

```
LCD CMD 64
LCD DATA 4, 14, 14, 14, 31, 0, 4, 0
```

(bijv. in de preamble) niet elke keer dat je de :

```
LCD 1, 1, CHR$(0)
```

Voor LCDI2C 16x2 of 20x4 scherm als volgt:

```
LCDI2C_Init &H3F, 1, 0, 2, 3 ' I2CAddr=&H20=20x4, &H38=16x2, &3F=16x2.
LCDI2C_CMD(64) ' Aangepast teken Chr$(0) is CMD(64) Reg(64-71)
LCDI2C_DATA(7) : LCDI2C_DATA(5) : LCDI2C_DATA(7) : LCDI2C_DATA(0) ' Byte 1 - 4
LCDI2C_DATA(0) : LCDI2C_DATA(0) : LCDI2C_DATA(0) : LCDI2C_DATA(0) ' Byte 5 - 8
LCDI2C 1, 1, "BuitenT:R 18.3 C" ' LCDI2C Line, Pos, Data$
LCDI2C 1,15, Chr$(0) ' Zet graden teken op Line 1, Pos=15, Chr$(0)
```

Zie ook subroutines LCDI2C.

Alle hardware I/O functies RP2040 en Chip pin nummers

Serial1: RX,TX Com1	SPI : RX,TX,SCK,CSn	I2C : SDA,SCL
RX : GP01,GP13,GP17,GP29	RX : GP00,GP04,GP16,GP20	SDA : GP0,GP4,GP08,GP12,GP16,GP20,GP28
TX : GP00,GP12,GP16,GP28	TX : GP03,GP07,GP19,GP23	SCL : GP1,GP5,GP09,GP13,GP17,GP21,GP29
CTS:GP02,GP14,GP18	CSn : GP01,GP05,GP17,GP21	I2C2 : SDA,SCL
RTS:GP03,GP15,GP19	SCK : GP02,GP06,GP18,GP22	SDA : GP2,GP6,GP10,GP14,GP18,GP22,GP26
Serial2: RX,TX COM2	SPI2: RX,TX,SCK,CSn	SCL : GP3,GP7,GP11,GP15,GP19,GP23,GP27
RX : GP05,GP09,GP21,GP25	RX : GP08,GP12,GP24,GP28	PWM : PWM0A-PWM7A en PWM0B-PWM7B
TX : GP04,GP08,GP20,GP24	TX : GP11,GP15,GP27	PWM0A: GP00,GP16
CTS:GP06,GP10,GP22,GP26	CSn : GP09,GP13,GP25,GP29	PWM0B: GP01,GP17
RTS:GP07,GP11,GP23,GP27	SCK : GP10,GP14,GP26	PWM1A: GP02,GP18
ADC: 12 Bit CH:0-4	GPIO:GP00-GP29	PWM1B: GP03,GP19
AD0:GP26 AD3:GP29=SYS	PIO0:GP00-GP29	PWM2A: GP04,GP20
AD1:GP27 AD4: TEMP	PIO1:GP00-GP29	PWM2B: GP05,GP21
AD2:GP28	RX : MasterInputSlaveOutput	PWM3A: GP06,GP22
	TX : MasterOutputSlaveInput	PWM3B: GP07,GP23
		PWM4A: GP08,GP24
		PWM4B: GP09,GP25
		PWM5A: GP10,GP26
		PWM5B: GP11,GP27
		PWM6A: GP12,GP28
		PWM6B: GP13,GP29
		PWM7A: GP14
		PWM7B: GP15

Pin nummers Raspberry Pi Pico en Chip pinnen

Chip Pin	PCB Pin	Chip Pin	PCB Pin
GP00	= 01	GP15	= 20
GP01	= 02	GP16	= 21
GP02	= 04	GP17	= 22
GP03	= 05	GP18	= 24
GP04	= 06	GP19	= 25
GP05	= 07	GP20	= 26
GP06	= 09	GP21	= 27
GP07	= 10	GP22	= 29
GP08	= 11	GP23	= 41 DOUT
GP09	= 12	GP24	= 42 DIN
GP10	= 14	GP25	= 43 HEATBEAT
GP11	= 15	GP26	= 31 ADC0
GP12	= 16	GP27	= 32 ADC1
GP13	= 17	GP28	= 34 ADC2
GP14	= 19	GP29	= 44 ADC3 VSYS/3

