

Bouw je eigen weerstation: deel 2

De I2C bus

De I2C bus (spreek uit: ie kwadraat cee) is een manier om apparaten en sensors op een eenvoudige manier aan elkaar te koppelen. Het woord bus betekent hier de draden die de verbinding vormen.

Als oefening gaan we een LCD aan de Arduino koppelen via de I2C bus. Hiervoor nemen we een LCD van 4 regels met 20 tekens per regel ook wel een LCD 20 x 4 genoemd. Aan de achterkant van deze LCD zit een speciaal printje om aan de I2C bus te kunnen koppelen.



LCD 20 x 4

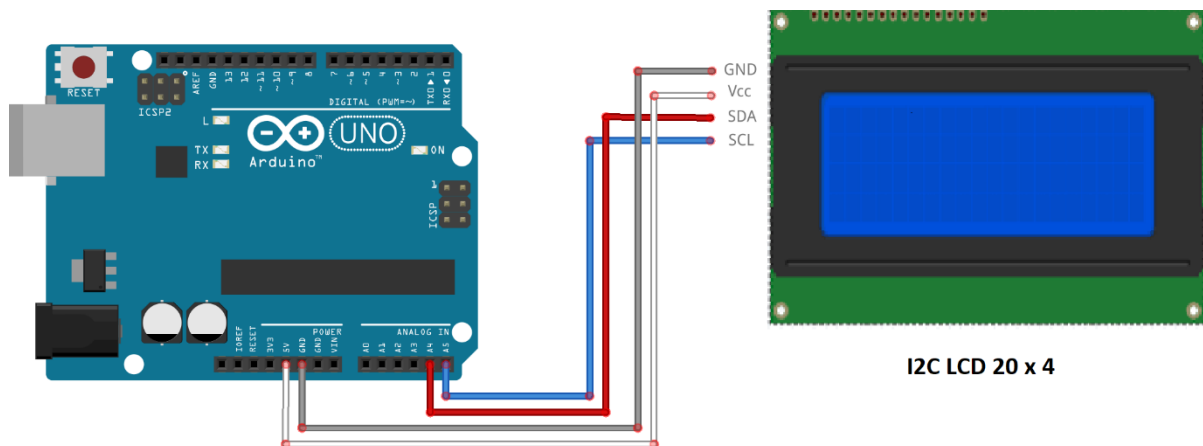


I2C printje



LCD met I2C printje

Aansluiten op de Arduino



I2C LCD 20 x 4

Programma om het I2C LCD te testen

In het programma hebben we 2 nieuwe zaken nodig:

- De I2C bibliotheek voor de Arduino deze heet 'Wire' en zit standaard in de IDE
- De I2C bibliotheek voor het LCD. Laad deze via schets -> bibliotheek gebruiken -> etc. etc. Zoeken op I2C LCD en de bibliotheek LiquidCrystal I2C van Frank de Brabander installeren.

De Arduino Uno gebruikt altijd de aansluitingen A4 en A5 voor de I2C bus. We maken nu een eenvoudig Arduino programma om het LCD te testen door op elke regel hiervan iets te schrijven.

De code weerstation 2a

```
/*weerstation_2a
*I2C 20 x 4 LCD scherm aansturing.
* LCD GND --> GND Arduino
* LCD VCC --> 5V Arduino
* LCD SDA --> A4 Arduino
* LCD SCL --> A5 Arduino
*I2C aansluitingen van Arduino Uno: A4 (SDA), A5 (SCL)
*/

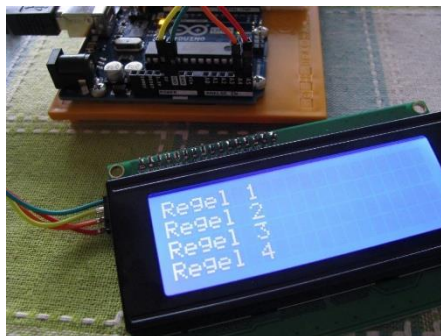
#include <Wire.h>           // Bibliotheek voor communicatie met I2C
#include <LiquidCrystal_I2C.h> // Bibliotheek voor het LCD scherm

LiquidCrystal_I2C lcd(0x27, 20, 4); // start LCD scherm

void setup() {
  lcd.init();           // initialiseer het LCD scherm
  lcd.backlight();     // zet de backlight aan
}

void loop() {
  lcd.clear();         // wis het scherm
  delay(1000);        // pauzeer 1 seconde
  lcd.setCursor(0, 0); // zet de cursor op positie 1, regel 1
  lcd.print ("Regel 1"); // schrijf op scherm
  delay(500);
  lcd.setCursor(0, 1); // zet de cursor op positie 1, regel 2
  lcd.print ("Regel 2"); // schrijf op scherm
  delay(500);
  lcd.setCursor(0, 2); // zet de cursor op positie 1, regel 3
  lcd.print ("Regel 3"); // schrijf op scherm
  delay(500);
  lcd.setCursor(0, 3); // zet de cursor op positie 1, regel 4
  lcd.print ("Regel 4"); // schrijf op scherm
  delay(500);
}
```

Als alles goed gegaan is zie je op het scherm:



...maar misschien zie je helemaal niets, terwijl je alles toch goed gedaan hebt, draai dan voorzichtig met een kleine schroevendraaier aan het blauwe potmetertje op het I2C printje van het LCD.

De Luchtdruk sensor

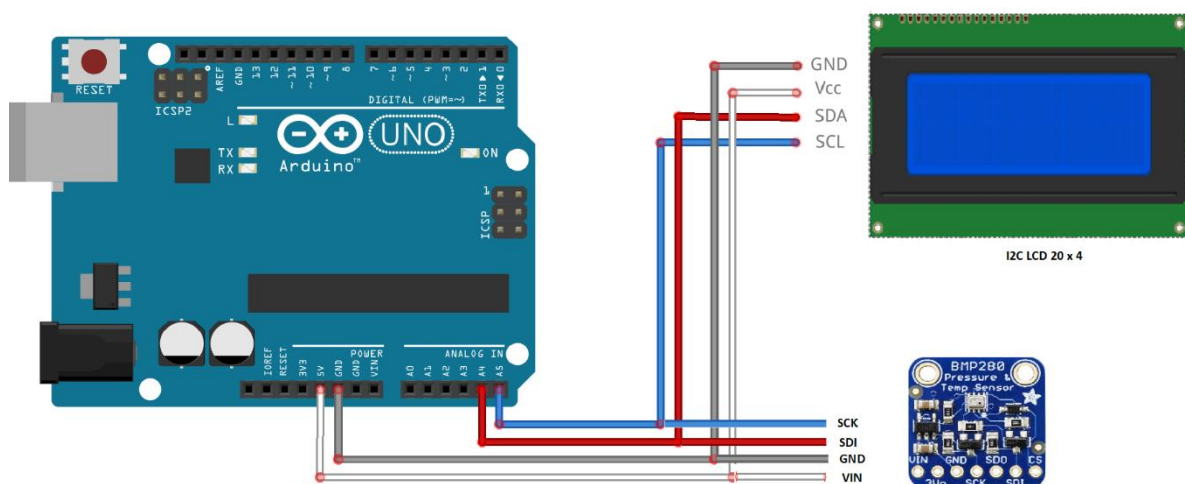
Een mooie en ook heel geschikte sensor om de luchtdruk te meten is de BMP280. Het afgebeelde printje van Adafruit gebruikt deze sensor en heeft bovendien nog een aantal componenten op het printje om deze eenvoudig op de Arduino te kunnen aansluiten. Deze sensor gebruikt ook de I2C bus om met de Arduino te communiceren.



Het Adafruit printje met de BMP280 luchtdruk sensor

Aansluiten van het LCD en de BMP280 op de Arduino

Gebruik het witte montage bordje om de verbindingen te maken!



Het uitlezen van de druksensor

Om de sensor in een programma uit te kunnen lezen moeten we de Adafruit BMP280 bibliotheek in de IDE laden (schets → bibliotheek gebruiken etc. etc.)

De code weerstation 2b

```
/*weerstation_2b  
* luchtdruk meten en de meetwaarde op het LCD plaatsen  
* I2C voor LCD en BMP 280 sensor  
*  
* LCD GND en BMP GND --> GND Arduino  
* LCD VCC en BMP Vin --> 5V Arduino  
* LCD SDA en BMP SDI --> A4 Arduino
```

```

* LCD SCL en BMP SCK --> A5 Arduino
*
* I2C aansluitingen van Arduino Uno: A4 (SDA), A5 (SCL)
*/

#include <Wire.h> // Bibliotheek voor communicatie met I2C
#include <LiquidCrystal_I2C.h> // Bibliotheek voor het LCD scherm
#include <Adafruit_BMP280.h> // Bibliotheek voor de BMP 280 sensor

LiquidCrystal_I2C lcd(0x27, 20, 4); // start LCD scherm en lcd is gebruiksnaam
Adafruit_BMP280 bmp; //bmp is nu de gebruiksnaam voor de druk sensor

float luchtdruk; // variabele om de luchtdruk op te slaan

void setup() {
  lcd.init(); // initialiseren van het LCD scherm
  lcd.backlight(); // zet de backlight aan
  bmp.begin(); // start de bmp sensor
}

void loop() {
  lcd.clear(); // wis het scherm
  delay(1000); // pauzeer 1 seconde
  lcd.setCursor(0, 0); // zet de cursor op positie 1, regel 1
  lcd.print ("Regel 1"); // schrijf op scherm
  delay(500);
  lcd.setCursor(0, 1); // zet de cursor op positie 1, regel 2
  lcd.print ("Regel 2"); // schrijf op scherm
  delay(500);
  luchtdruk = bmp.readPressure()/100.0;
  lcd.setCursor(0, 2); // zet de cursor op positie 1, regel 3
  lcd.print ("Luchtdruk ");
  lcd.print (luchtdruk,1); // schrijf op scherm
  lcd.print (" hPa");
  delay(500);
  lcd.setCursor(0, 3); // zet de cursor op positie 1, regel 4
  lcd.print ("Regel 4"); // schrijf op scherm
  delay(1000);
}

```

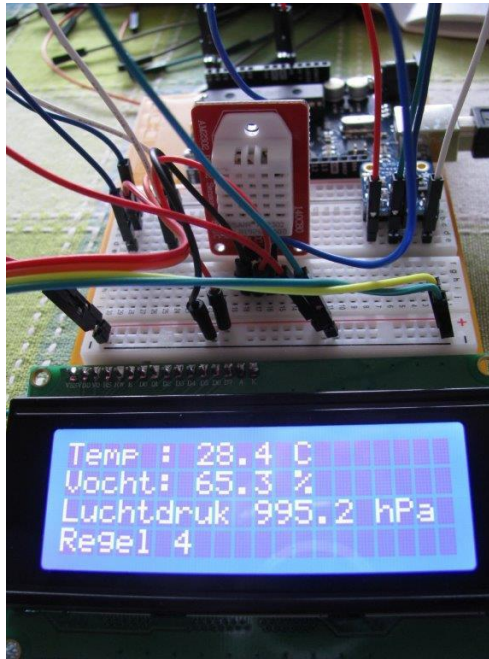
Op regel 3 wordt de luchtdruk geschreven.



Nu zelf aan het werk

Combineer weerstation deel 1 (temperatuur en luchtvochtigheid meting) met de luchtdruk meting in de vorige opgave.

De uitkomst kan zijn als in het volgende plaatje.



Hints:

- Sluit de DHT22 sensor aan zoals aangegeven is in weerstation deel 1
- Combineer de sketches

Kom je er niet uit kijk dan hieronder hoe je dit zou kunnen doen.

Code weerstation 2c

```
/* sketch weerstation_2c
 * Temperatuur, luchtvochtigheid en luchtdruk meten en de meetwaarde op het LCD plaatsen
 * I2C voor LCD en BMP 280 sensor
 *
 * LCD GND en BMP GND --> GND Arduino
 * LCD VCC en BMP Vin --> 5V Arduino
 * LCD SDA en BMP SDI --> A4 Arduino
 * LCD SCL en BMP SCK --> A5 Arduino
 *
 * I2C aansluitingen van Arduino Uno:A4 (SDA), A5 (SCL)
 *
 * DHT22 voor temperatuur en vochtigheid
 * ARDUINO pen 9 voor data van DHT22
 */

#include <Wire.h> // Bibliotheek voor communicatie met I2C
#include <LiquidCrystal_I2C.h> // Bibliotheek voor het I2C LCD scherm
#include <Adafruit_BMP280.h> // Bibliotheek voor de BMP 280 sensor
#include <DHT.h> // Bibliotheek voor de DHT sensor
```

```

LiquidCrystal_I2C lcd(0x27, 20, 4); // start LCD scherm en lcd is gebruiksnaam
Adafruit_BMP280 bmp; // bmp is nu de gebruiksnaam voor de druk sensor
DHT sensor_dht(9, DHT22); // onze sensor heet nu: sensor_dht

float luchtdruk, temperatuur, vochtigheid; // variabelen

void setup() {
  lcd.init(); // initialiseren van het LCD scherm
  lcd.backlight(); // zet het backlight aan
  bmp.begin(); // start de bmp sensor
  sensor_dht.begin(); // start de dht sensor
}

void loop() {
  delay(2000);
  lcd.clear(); // wis het scherm
  delay(500);
  //lees de temperatuur, vochtigheid en luchtdruk uit en bewaar die in de variabelen
  temperatuur = sensor_dht.readTemperature();
  vochtigheid = sensor_dht.readHumidity();
  luchtdruk = bmp.readPressure()/100.0;
  lcd.setCursor(0, 0); // zet de cursor op positie 1, regel 1
  lcd.print ("Temp : "); lcd.print(temperatuur,1); lcd.print (" C ");
  delay(500);
  lcd.setCursor(0, 1); // zet de cursor op positie 1, regel 2
  lcd.print ("Vocht: "); lcd.print (vochtigheid,1); lcd.print (" % ");
  delay(500);
  lcd.setCursor(0, 2); // zet de cursor op positie 1, regel 3
  lcd.print ("Luchtdruk "); lcd.print (luchtdruk,1); lcd.print (" hPa");
  delay(500);
  lcd.setCursor(0, 3); // zet de cursor op positie 1, regel 4
  lcd.print ("Regel 4"); // schrijf op scherm
  delay(1000);
}

```