



SENSOREN EN INTERFACING

MASTER INGEBEDDE SYSTEMEN 2018–2019

Weerstation

Groep 12: *Jonas Van der donckt* *Mattijs Lootens*

Docenten: Brecht Willems, Jeroen Missine

20 oktober 2018

Inhoudsopgave

1	Inleiding	2
2	Materiaal en methode	2
2.1	Arduino Uno	2
2.2	LDR	2
2.2.1	Werking	2
2.2.2	Principe	2
2.2.3	Werkinggebied	2
2.2.4	Belangrijke informatie bij het implementeren	2
2.2.5	I/O uitleg	3
2.2.6	Moeilijkheidsgraad	3
2.3	DHT	3
2.3.1	Werking	3
2.3.2	Principe	3
2.3.3	Werkinggebied	4
2.3.4	Belangrijke informatie bij het implementeren	4
2.3.5	I/O uitleg	4
2.3.6	Moeilijkheidsgraad	5
2.4	MQ-2	5
2.4.1	Werking	5
2.4.2	Principe	5
2.4.3	Werkinggebied	5
2.4.4	Belangrijke informatie bij het implementeren	6
2.4.5	I/O uitleg	6
2.4.6	Moeilijkheidsgraad	6
2.5	BMP085	6
2.5.1	Werking	6
2.5.2	Principe	6
2.5.3	Werkinggebied	7
2.5.4	Belangrijke informatie bij het implementeren	7
2.5.5	I/O uitleg	7
2.5.6	Moeilijkheidsgraad	7
2.6	Rotary encoder	7
2.6.1	Werking	7
2.6.2	Principe	8
2.6.3	Werkinggebied	8
2.6.4	Belangrijke informatie bij het implementeren	8
2.6.5	I/O uitleg	9
2.6.6	Moeilijkheidsgraad	9
3	Totaalpakket	9
3.1	ESP8266	9
3.1.1	Interfacing met Arduino	10
3.2	Flowchart	11
3.3	Code principes	12
4	Besluit	12

1 Inleiding

Het weer beïnvloedt het dagelijkse leven. Van mooi weer wordt je beter gezind vergeleken met dagen waarop het door en door regent. Daarom is het nodig om te kunnen meten welke parameters de weersituatie kenmerkt om hiermee eventueel de komende weersituatie te voorspellen. Dit kan gedaan worden in een weerstation. Omdat er niet overal evenveel plaats is om een volledig weerstation te bouwen, kan het interessant zijn om microweerstations te creëren dat bij iedereen thuis kan staan om metingen te doen. Voor dit labo wordt dergelijk station geïmplementeerd in met behulp van een Arduino microcontroller.

2 Materiaal en methode

2.1 Arduino Uno

Het microcontrollerbord dat wordt gebruikt is de Arduino Uno. Hiermee worden alle sensorwaarden ingelezen, eventueel verwerkt en de actuators aangepast. Het bord wordt geprogrammeerd in de CLion IDE aan de hand van de Platformio plugin.

De gemeten waarden worden via een ESP8266 microcontroller, die een wifi module bevat, naar de Thingspeak™ API verstuurd. Dit is een IoT analyse service, waarmee de waarden gevisualiseerd kunnen worden.

2.2 LDR

2.2.1 Werking

Een LDR of light-dependent resistor is niets meer dan een weerstand waarvan de waarde afhankelijk is van de hoeveelheid licht die erop valt.

2.2.2 Principe

Deze weerstand werkt aan de hand van het foto-elektrisch effect. Door licht krijgen elektronen in de valentieband genoeg energie om te exiteren naar de geleidingsband. Hierdoor stijgt de elektrische geleiding en daalt de weerstand van het component.

2.2.3 Werkingsgebied

De gebruikte LDR heeft bij daglicht een weerstand van $1.4k\Omega$. Als er een lamp van een gsm op schijnt daalt de weerstand tot een waarde van ongeveer 500Ω . De gemeten waarden variëren van 100 lux naar ongeveer 7000 lux.

2.2.4 Belangrijke informatie bij het implementeren

De gelezen analoge waarde daalt naarmate dat de LDR meer licht opneemt. Deze meetwaarde is dus omgekeerd evenredig met de te meten verlichtingssterkte. Hier moet rekening mee worden gehouden om tot een juiste waarde te komen. De juiste factor moet gevonden worden aan de hand van de datasheet of empirisch onderzoek om de spanning om te zetten naar een lux waarde.

2.2.5 I/O uitleg

De LDR moet aangesloten worden aan een analoge pin van de Arduino om een juiste waarde te bekomen. De sterkte van het signaal kan niet digitaal bepaald worden om duidelijke redenen.

2.2.6 Moeilijkheidsgraad

De LDR wordt verbonden met de Arduino via een spanningsdeler-configuratie. Er moet dus opgelet worden dat de schakeling in orde is. Het gebruiken van een LDR is overigens redelijk vanzelfsprekend. Het grote nadeel van deze sensor was, dat er geen corresponderend datasheet gevonden kon worden. Hierdoor kunnen er geen concrete uitspraken gedaan worden omtrent het bereik/de nauwkeurigheid enz...

2.3 DHT

2.3.1 Werking

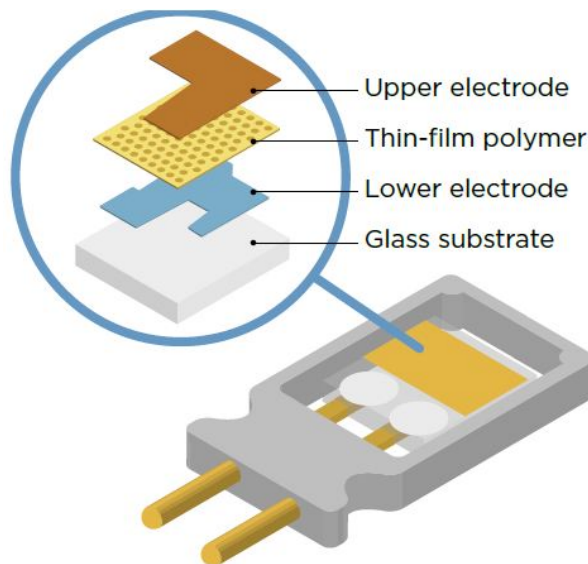
Een Digital Humidity and Temperature sensor is een component waarmee zowel temperatuur als vochtigheidsgraad gemeten kan worden. In dit geval wordt een DHT11 gebruikt in plaats van de preciezere DHT22.

2.3.2 Principe

In een DHT zitten twee aparte sensorcomponenten verwerkt: een temperatuursensor (of thermistor) en een vochtigheidsmetend component.

De thermistor werkt gelijkaardig zoals de eerder besproken LDR, maar is afhankelijk van temperatuur. Dit is een NTC *negative temperature coefficient* thermistor, wat betekent dat de weerstand omgekeerd evenredig is met de temperatuur. Deze sensoren worden gemaakt door het insmelten van halfgeleidermaterialen. Door de temperatuur van deze halfgeleiders te laten stijgen, groeit het aantal elektronen in de geleidingsband, waardoor het materiaal beter gaat geleiden.

Het vochtigheidsmetend component bestaat uit twee electrodes met tussenin een vochthoudend substraat. Een visualisatie van deze opstelling is te zien in figuur 1. Zo verandert de geleidbaarheid van het substraat gelijk met de vochtigheid. Deze weerstand wordt dan door het geïntegreerd circuit verwerkt zodat het makkelijk kan gestuurd worden naar de microcontroller die de metingen uitvoert.



Figuur 1: Vochtigheidsmetend component

2.3.3 Werkingsgebied

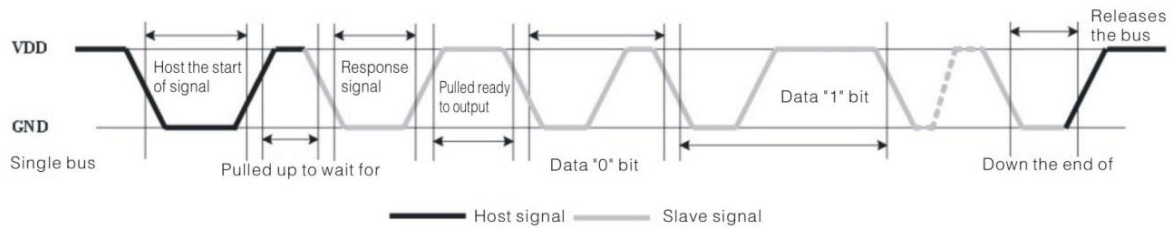
De DHT11 heeft een nauwkeurigheid van $\pm 2^{\circ}\text{C}$ in temperatuur en $\pm 5\%$ in vochtigheid. Voor de temperatuursensor kunnen slechts waarden binnen de range van 0°C - 50°C gemeten worden. Het bereik voor de vochtigheid mag variëren tussen 20% - 80%.

2.3.4 Belangrijke informatie bij het implementeren

De DHT sensor wordt het best geïmplementeerd met behulp van een DHT bibliotheek. Er wordt gebruik gemaakt van de DHTLib die te vinden is op de arduino playground. Deze kan gebruikt worden voor zowel DHT11, -21, -22, -33 en -44. Als enkel de DHT11 gebruikt wordt is het interessanter om DHT11Lib te gebruiken aangezien deze compacter is, maar in dit geval wordt voor gemak toch DHTLib gebruikt. Om de sensor in te lezen wordt `DHT.read11` gebruikt en vervolgens kan toegang tot de waarden via `DHT.temperature` en `DHT.humidity` verkregen worden. Er moet wel elke loop opnieuw gelezen worden, anders zullen de waarden niet updaten.

2.3.5 I/O uitleg

De DHT sensor heeft zijn eigen protocol om data te versturen op een digitale seriële manier. Het timing schema is te zien in figuur 2. Hierop is te zien dat er precieze timing vereist is om de juiste data te versturen. Gelukkig moet hier niet veel rekening mee worden gehouden aangezien de bibliotheek dit zelf doet.



Figuur 2: Timing schema voor DHT data transport.

2.3.6 Moeilijkheidsgraad

Zonder de DHTLib zou het implementeren van deze sensor een ware uitdaging zijn. Gelukkig kan een bibliotheek dit vereenvoudigen waardoor het gebruiken van deze sensor een stuk eenvoudiger wordt. Er moet enkel opgelet worden of de juiste DHT sensor gebruikt wordt.

2.4 MQ-2

2.4.1 Werking

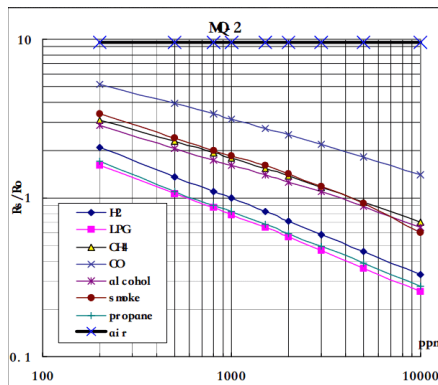
De MQ-2 is een rookdetecterende sensor. Deze sensor is gevoelig voor zowel rook als andere ontvlambare gassen, zoals bijvoorbeeld butaan, propaan of methaan. De output spanning is afhankelijk van de concentratie gas dat opgenomen wordt.

2.4.2 Principe

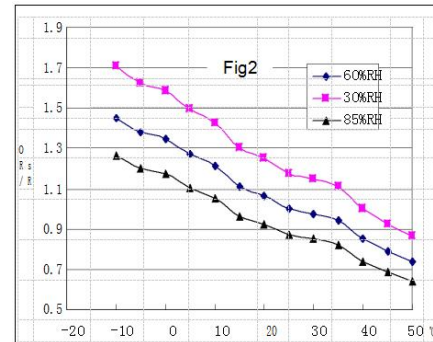
De MQ-2 heeft een elektrochemische sensor waarvan de weerstand verandert naarmate de concentratie gas in de atmosfeer. Deze elektrochemische sensor werkt aan de hand van elektrodes. Een gas dringt binnen in de sensor, waardoor ze contact maakt met de elektrodes. Bij contact ontstaan chemische reacties, ofwel een oxidatie, ofwel een reductie, afhankelijk van het soort gas. Door deze reactie ontstaat er een elektronenstroom. Aan de hand van een spanningsdeler en deze weerstand zal de uitgang-spanning dus variëren door de gedetecteerde gasconcentratie. Deze spanning kan door de microcontroller geïnterpreteerd worden en zo kan er een grens bepaald worden welke hoeveelheid rook gezien wordt als 'te veel' rook.

2.4.3 Werkingsgebied

De sensor detecteert gassen tussen concentraties van 300 en 10000 ppm. Naast de concentratie is de sensor ook gevoelig aan temperatuur en vochtigheid. De karakteristieken zijn te zien in figuren 3 en 4.



Figuur 3: Karakteristiek van de sensor bij verschillende gassen



Figuur 4: Karakteristiek van de sensor bij temperatuur en vochtigheid

2.4.4 Belangrijke informatie bij het implementeren

Deze sensor zal worden gebruikt om een soort brandalarm te implementeren. Omdat deze sensor nog niet gekalibreerd is, kan op zich niet een grens worden vastgesteld. Er moet empirisch bepaald worden vanaf welke sensorwaarde echte rook gedetecteerd wordt, en dan kan een grens vastgelegd worden. Als deze grens overschreden wordt, kan een alarm afgaan in de vorm van een buzzer en/of een lichtje.

2.4.5 I/O uitleg

De sensor werkt analoog, dus moet de output ook aan een analoge pin verbonden worden. Er kan ook een digitale output aan de microcontroller verbonden worden die hoog wordt als de gemeten waarde boven een grens gaat. Deze grens is in te stellen met de potentiometer op de sensor zelf.

2.4.6 Moeilijkheidsgraad

De sensor is redelijk simpel te gebruiken, verbinden en uitlezen. Er hoeft geen bibliotheek gebruikt te worden om de sensor werkende te krijgen.

2.5 BMP085

2.5.1 Werking

De BMP085 is een barometer en thermometer in één. Het is een I2C sensor, waardoor deze makkelijk aan te sluiten is. Aan de hand van de gemeten parameters kan ook de hoogte berekend worden, op voorwaarde dat rekening gehouden wordt met de luchtdruk op zeeniveau.

2.5.2 Principe

De BMP085 bestaat uit een piezo-resistieve sensor, een converter om analoge signalen om te zetten naar digitale, een control unit en een serial interface. De piezo-resistieve sensor is wat de luchtdruk meet. Het is een weerstand waarvan de geleidbaarheid afhankelijk is van druk of trek. Deze sensors worden vaak vervaardigd uit silicone, aangezien dit materiaal een groter piezo-elektrisch effect heeft dan bijvoorbeeld metaal. De druksensor bestaat uit dunne siliconen

vlakken. Het geïntegreerd circuit detecteert verandering in de weerstand waaruit de druk afgeleid kan worden. De BMP085 heeft een ingeboude E²PROM waar de kalibratie-data opgeslagen wordt. De sensor moet dus niet meer door de gebruiker gekalibreerd worden.

2.5.3 Werkingsgebied

Het werkingsgebied van deze sensor is te vinden in de datasheet. De druk mag variëren van 300 tot 1100 hPa. De sensor is operationeel bij een temperatuur tussen -40 en 85°C , maar volledige nauwkeurigheid kan slechts bereikt worden tussen 0 en 65°C . De nauwkeurigheid waarmee de sensor in deze ideale omgevingstoestand kan meten is ± 1.0 hPa en $\pm 1.0^{\circ}\text{C}$.

2.5.4 Belangrijke informatie bij het implementeren

Om de sensor data op een universele manier te genereren wordt er gebruik gemaakt van de Adafruit_Sensor bibliotheek. De berekeningen en timing schema's om de sensor te doen werken zijn allemaal beschikbaar in de datasheet, maar om dit allemaal te implementeren wordt veel tijd verloren. Het is efficiënter om de Adafruit_BMP085 Arduino bibliotheek te installeren en gebruiken. Eénmaal de BMP geïntialiseerd is met het Adafruit_BMP085.Unified commando, kan de gemeten data verkregen worden door een sensor event op te vragen en de gewenste parameters te weergeven, in dit geval pressure en/of temperature.

2.5.5 I/O uitleg

Omdat de sensor een ingebouwd I²C interface heeft moet deze aangesloten zijn aan de I²C clock en data pinnen, dit zijn respectievelijk pin 5 en 4. De sensor moet uiteraard nog verbonden worden aan een voeding en een grond. Voor deze sensor wordt de 3.3V voeding gebruikt. De BMP085 V2 heeft een regulator die 5V omzet naar 3.3V, maar voor de veiligheid wordt toch een spanningsbron van 3.3V aangesloten op de sensor.

2.5.6 Moeilijkheidsgraad

Doordat er opnieuw een bibliotheek gebruikt kan worden, ligt de moeilijkheid relatief laag. Deze Adafruit_BMP085 bibliotheek detecteert automatisch op welke pinnen de BMP085 aangesloten is. Dit vereenvoudigt het gebruiksgemak.

Er is ook een optie om de altitude weer te geven. Hiervoor moet de luchtdruk op zeeniveau bekend zijn. In de code is deze als een constante gedefinieerd. Aangezien dit in de realiteit varieert, is deze hoogte waarde niet representatief. Voor deze reden werd er ook gezocht naar een gratis API waar men de luchtdruk op zeeniveau kan verkrijgen. De wifi module zou dan deze API kunnen gebruiken om de huidige luchtdruk op zeeniveau te verkrijgen, wat als bijgevolg een betere voorspelling zal leveren. Openweathermap leek hiervoor de ideale oplossing. Na wat experimenteren met deze API, bleek dat deze niet zo nauwkeurig was. De luchtdruk bleef hetzelfde voor Kortrijk, Nieuwpoort, ... Aangezien dit eerder afwijkt van de scope van dit project, werd er uiteindelijk gekozen om het bij een constante luchtdruk waarde op zeeniveau te houden in de code (1020 hPa).

2.6 Rotary encoder

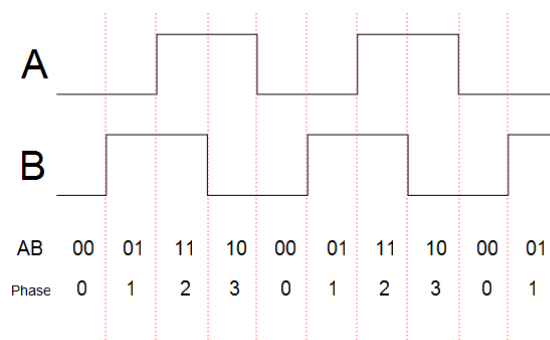
2.6.1 Werking

Een rotary encoder is een digitale sensor waarmee de hoek van een draaibare knop gemeten kan worden. Er zijn verschillende soorten rotary encoders, afhankelijk van hoe de output eruitziet

en op welke manier de rotatie gedetecteerd wordt. In dit geval wordt er met een incrementele rotary encoder gewerkt.

2.6.2 Principe

Door aan de knop te draaien worden er blokgolven gegenereerd. Dit gebeurt aan de hand van een schijf met gelijkmatig verspreide contact zones. Bij het ronddraaien komen twee pinnen in een bepaalde volgorde in contact met deze zones, waardoor er een spanning op de pinnen komt te staan. Zo ontstaan er op de pinnen twee gelijkaardige blokgolven die niet in fase staan. Voor de duidelijkheid is een visualisatie van de golfvormen te zien in figuur 5. De richting waarin de knop draait kan afgeleid worden uit de volgorde van de toestanden, weergegeven door AB. De volgorde waarop de blokgolven hoog of laag worden, wijst namelijk op de volgorde waarop de pinnen in contact komen met de zones. Op die manier kan de hoek waarin de knop staat geregistreerd worden.



Figuur 5: Golfvorm rotary encoder

2.6.3 Werkingsgebied

Omdat het draaien van de knop vooral in software wordt bijgehouden, hangt het werkingsgebied eigenlijk vooral af van de manier waarop deze geïmplementeerd wordt. Als enkel de stand van de knop wordt bijgehouden is de enige limiet de nauwkeurigheid. Als wordt bijgehouden hoe vaak de knop een volledige omwenteling heeft gemaakt zal er ook met het geheugen rekening gehouden moeten worden. De sensor heeft 40 verschillende toestanden waarin ze kan staan alvorens terug in de begintoestand (hoek 0°) te staan. Dit betekent dus dat met een nauwkeurigheid van $\pm 9^\circ$ gemeten kan worden. Let wel dat deze toestanden niet allemaal even stabiel zijn. De knop is ontworpen om stabiel te zijn op 20 van de 40 plaatsen.

2.6.4 Belangrijke informatie bij het implementeren

Omdat deze sensor zeer snel moet kunnen reageren op verandering in het signaal zal er met interrupts moeten worden gewerkt. Andere sensoren werken namelijk mogelijk met een delay, wat de werking van deze sensor helemaal in de war kan brengen. Daarom moet deze sensor ook op de juiste pinnen worden aangesloten. Voor het gebruikte bord, de Arduino Uno, zijn dit pin 2 en 3. De sensor heeft ook een ingebouwde button functionaliteit, maar deze wordt voor het weerstation toch achterwege gelaten.

2.6.5 I/O uitleg

De twee outputs zijn verbonden met de digitale interrupt pinnen. Deze worden zo geprogrammeerd dat bij een verandering van signaal een vooraf gedefinieerde methode wordt opgeroepen. Op die manier kan de microcontroller zijn werking snel onderbreken om de stand van de rotary encoder bij te kunnen houden.

2.6.6 Moeilijkheidsgraad

De moeilijkheidsgraad bij het implementeren van deze encoder komen voor bij het samenwerken met andere functionaliteiten. Indien enkel een encoder geïmplementeerd moet worden is er geen probleem en kan alles in een loop geprogrammeerd worden, maar zodra een andere sensor of actuator geïmplementeerd wordt die de werking vertraagd aan de hand van bijvoorbeeld een delay, moet er met interrupts gewerkt worden.

3 Totaalpakket

Het ontworpen weerstation meet de waarden van de zojuist besproken sensoren op en stuurt deze door naar een Thingspeak channel. Hiervoor moet de Arduino Uno een netwerk verbinding hebben. Dit wordt gerealiseerd door gebruik te maken van de ESP8266 WiFi-module. Deze zal besproken worden in de volgende paragraaf.

Volgende tabel geeft een overzicht van de gebruikte sensoren en hun betrekking tot het weerstation.

Sensor	Nut
BMP085	Meet de luchtdruk (kPa), temperatuur en hoogte, wordt doorgestuurd naar Thingspeak
DHT11	Meet de luchtvochtigheid (%), wordt doorgestuurd naar Thingspeak
LDR	Meet de lichtintensiteit (Lux), wordt doorgestuurd naar Thingspeak
MQ2	Meet de gasconcentratie, wordt doorgestuurd naar Thingspeak
Rotary encoder	De hoek wordt gemeten en bepaalt de update frequentie naar Thingspeak.

Voor dit project moesten er ook een paar verplichte actuatoren gebruikt worden.

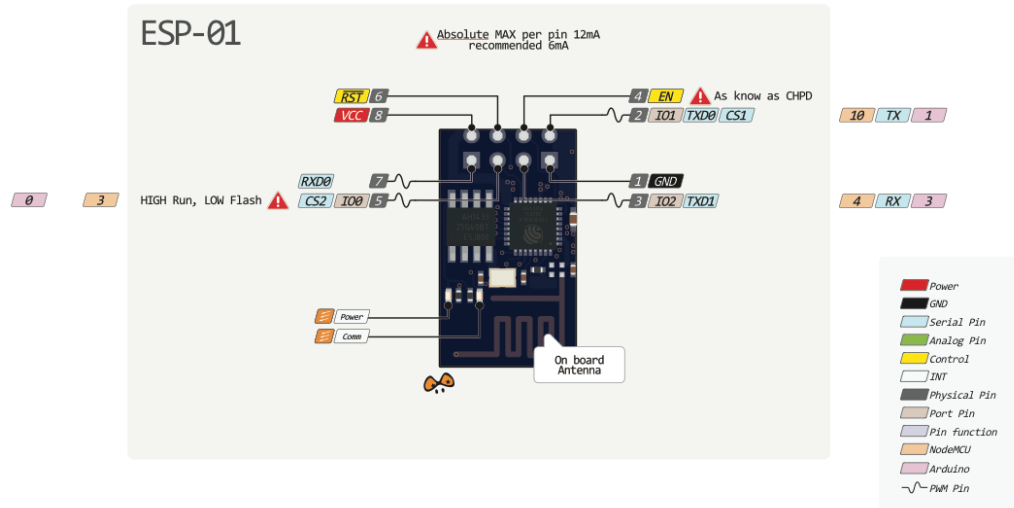
Actuator	Nut
(RGB) LED	Geeft de status van de ESP 8266 module weer
Buzzer	Gaat af indien een gasconcentratie boven een bepaalde threshold gemeten wordt.

3.1 ESP8266

ESP8266 is een Wi-Fi microcontroller, geproduceerd door Espressif Systems. Aangezien dit dus een microcontroller op zich is, bestaat er de optie om op de ESP8266 direct de sensor waarden uit te lezen en door te sturen. Aangezien het aantal pinnen van de gebruikte ESP module beperkt waren (slechts 3 GPIO pinnen), is ervoor gekozen om deze ESP8266 met een Arduino Uno te interfaceren. Dit werd gedaan door gebruik te maken van de AT commando's.

ESP8266/PINOUT

⚠ All ESP8266 in/outs are NOT 5V tolerant!

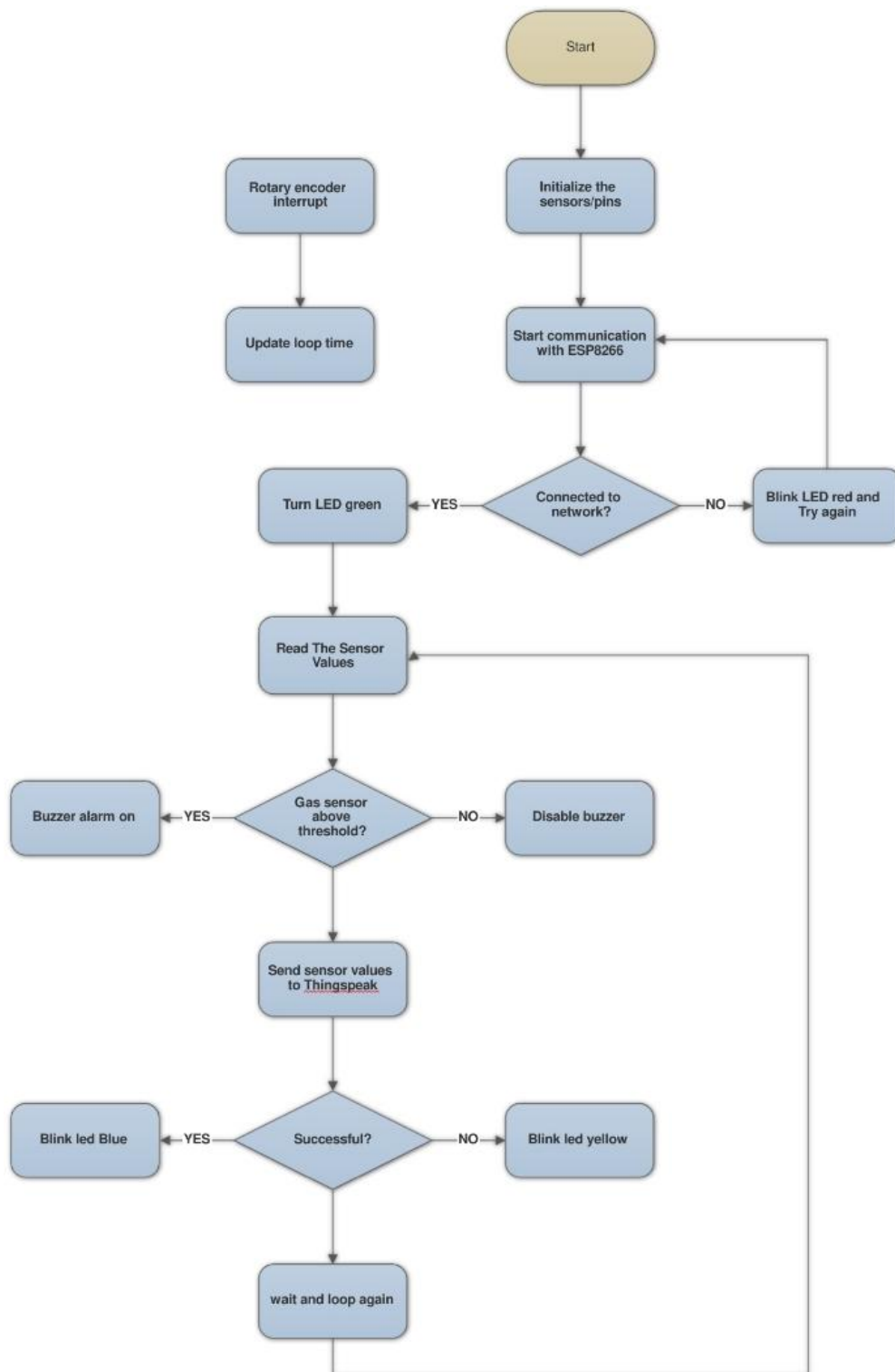


Figuur 6: ESP8266 microcontroller

3.1.1 Interfacing met Arduino

De ESP8266 werd eerst geflashed zodat hij op 9600 baud zou communiceren. Op verschillende fora werd er namelijk gezegd dat de standaard 115200 baud onstabiel is. Aangezien de ESP8266 op 3.3 Volt werkt, werd deze apart gevoed via een buck-converter. Er werd geopteerd om niet de 3.3V voeding van de Arduino te gebruiken omdat de ESP soms meer stroom (300mA) kan trekken als hij aan het versturen is. Logischerwijze moet er ook een spanningsdeler gemaakt worden voor de (RX en TX) signalen van Arduino naar de ESP te her schalen.

3.2 Flowchart

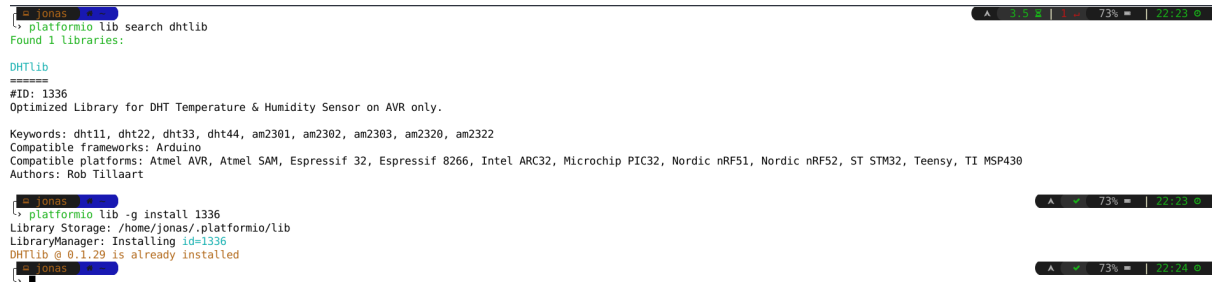


Figuur 7: Flowchart

3.3 Code principes

Voor de code werd er gebruik gemaakt van PlatformIO. Dit is een platform/tool die heel vaak gebruikt wordt voor embedded development. Het grote voordeel aan PlatformIO is de flexibiliteit. Aangezien de code open source is, spoort dit aan om veel meer plug-ins rond dit platform te bouwen. Hierdoor geven de meeste IDE's (Atom, Visual Studio Code, CLion, ...) support aan het PlatformIO platform.

De code werd in c++ geschreven. Zoals er in de vorige paragrafen beschreven, werd er gebruik gemaakt van bestaande bibliotheken. Deze konden eenvoudig geïnstalleerd worden via PlatformIO in de commandline.



```
j@jonas ~$ platformio lib search dhtlib
Found 1 libraries:

DHTLib
=====
#ID: 1336
Optimized Library for DHT Temperature & Humidity Sensor on AVR only.

Keywords: dht11, dht22, dht33, dht44, am2301, am2302, am2303, am2320, am2322
Compatible frameworks: Arduino
Compatible platforms: Atmel AVR, Atmel SAM, Espressif 32, Espressif 8266, Intel ARC32, Microchip PIC32, Nordic nRF51, Nordic nRF52, ST STM32, Teensy, TI MSP430
Authors: Rob Tillaart

j@jonas ~$ platformio lib -g install 1336
Library Storage: /home/jonas/.platformio/lib
LibraryManager: Installing id=1336
DHTLib @ 0.1.29 is already installed
```

Figuur 8: Installeren van een bibliotheek

De code kan gevonden worden op deze GitHub repository.

4 Besluit

Dit project was een goede inleiding tot het aansturen en onderzoeken van verschillende sensoren. Er werd geleerd over de verschillende technieken hoe de sensoren een fysische eigenschap omzetten naar een corresponderend elektrisch signaal. Desondanks waren er ook enkele hekel punten.

Doordat de aangeboden sensoren uit een goedkope sensor kit kwamen, was er geen officiële documentatie van de meeste sensoren. Hierdoor werd er een empirische formule bepaald voor het elektrisch signaal van de LDR naar Lux om te zetten.

De interfacing met de ESP8266 verliep ook niet echt vlot. Initieel had deze ESP een baudrate van 115200 baud, wat ervoor zorgde dat de communicatie onstabiel was. De 3.3V voeding van de Arduino Uno, gaf niet voldoende stroom voor de ESP. Er werd dus een aparte voeding van 3.3V gebruikt.

Tenslotte willen we prof. dr. Willems bedanken voor de uitstekende ondersteuning in de labo sessies.