

# Verbeterde firmware GEN-node



## Afstudeerverslag

*5 juni 2020, Gouda*

**Opdrachtgever**

Olaf Peters

**Auteur**

Jeremy Hut

*De Haagse Hogeschool, opleiding HBO-ICT*



**DE HAAGSE**  
HOGESCHOOL

# Referaat

Jeremy Hut, “Verbeterde firmware GEN-node”. Afstudeerverslag HBO-ICT, De Haagse Hogeschool 2020.

Dit afstudeerverslag toont het proces van het project waarin de firmware van de GEN-node (een sensornode) wordt verbeterd. Dit wordt gedaan aan de hand van ondersteuning voor analoge sensoren, ondersteuning voor geselecteerde luchtkwaliteitssensoren en een verlaging van het energieverbruik.

# Inhoudsopgave

<b>1. Inleiding</b>	<b>1</b>
<b>2. Organisatie Technolution</b>	<b>2</b>
2.1 Organisatiestructuur	2
2.2 Probleemdomein	3
<b>3. Opdracht</b>	<b>4</b>
3.1 Aanleiding	4
3.2 Probleemstelling	4
3.3 Doelstelling	4
3.4 Resultaat	4
3.5 Randvoorwaarden en afbakening	4
3.6 Aanpak	5
<b>4. Oriëntatiefase</b>	<b>7</b>
4.1 Verhelderen opdracht	7
4.2 Risico's analyseren	8
4.3 Planning	10
4.3.1 Methode	10
4.3.2 Fasering	11
<b>5. Analysefase</b>	<b>13</b>
5.1 Huidige GEN-node	13
5.1.1 Sense2Grow-netwerk	13
5.1.2 Printed Circuit Board (PCB)	14
5.1.3 Software	16
5.2 Onderzoek gewenste firmware GEN-node	19
5.2.1 Analoge sensoren universeel ondersteunen	19
5.2.2 Geselecteerde luchtkwaliteitssensoren ondersteunen	21
5.2.3 Energieverbruik verbeteren	22
5.3 Requirements achterhalen	24
5.3.1 Functionele requirements	24
5.3.2 Niet-functionele requirements	25
5.4 Systeemeisen vaststellen	25
<b>6. Iteratie 1: ondersteunen analoge sensoren</b>	<b>26</b>
6.1 Ontwerpfase	26
6.1.1 Ondersteuning analoge sensoren	27
6.1.2 Ondersteuning gewenste sensoren	29
6.2 Ontwikkelfase	33
6.2.1 Applicatie analog sensor	33
6.2.2 Applicatie multi sensor	36

6.3 Testfase	38
<b>7. Iteratie 2: Luchtkwaliteit</b>	<b>41</b>
7.1 Ontwerpfase	41
7.1.1 Alphasense gassensoren	41
7.1.2 PT500	43
7.2 Ontwikkelfase	44
7.3 Testfase	45
<b>8. Iteratie 3: Energieverbruik</b>	<b>47</b>
8.1 Ontwerpfase	47
8.2 Ontwikkelfase	52
8.2.1 Energie	52
8.2.2 Drivers	52
8.3 Testfase	53
<b>9. Adviesfase</b>	<b>55</b>
<b>10. Conclusie</b>	<b>57</b>
<b>11. Evaluatie</b>	<b>59</b>
11.1 Kwaliteit van producten	59
11.2 Procesevaluatie	60
<b>12. Aantonen beroepstaken</b>	<b>63</b>
<b>13. Verklarende woordenlijst</b>	<b>65</b>
<b>Literatuur</b>	<b>66</b>
<b>Bijlagen</b>	
A. Plan van Aanpak	
B. Analyserapport	
C. Ontwerprapport	
D. Testrapport	
E. Adviesrapport	

# 1. Inleiding

Van oorsprong is Technolution een bedrijf dat zich bezighoudt met technische automatisering, waaronder binnen de agrarische sector. En juist voor deze sector is het Sense2Grow-netwerk ontstaan. Sense2Grow is een sensornetwerk met bijbehorend platform. Door in bijvoorbeeld kassen sensoren op te kunnen hangen en de meetgegevens te verzamelen kan er gestuurd worden op een optimalere groei van de gewassen. Omdat er met zo'n IoT-netwerk als Sense2Grow meer mogelijkheden zijn dan enkel de agrarische sector is het gewenst dat dit systeem toegepast kan worden in andere omgevingen.



» Afbeelding 1.1, GEN-node

Het Sense2Grow-netwerk maakt gebruik van zogeheten GEN-nodes. Dit zijn sensornodes (zie afbeelding 1.1) die een aangesloten sensor uitleest. Vervolgens wordt dit draadloos verstuurd door middel van LoRa. In dit project wordt de firmware verbeterd van de GEN-node.

In de eerste fase van dit afstudeerproject is er een Plan van Aanpak gemaakt. Hier is het doel opgenomen: “De afstudeeropdracht leidt tot verbeterde firmware voor een Proof-of-Concept-model GEN-node die ondersteuning heeft voor analoge sensoren, plus geselecteerde sensoren voor het meten van luchtkwaliteit waarbij het energieverbruik verlaagd is.”

In de tweede fase wordt het probleemdomein geanalyseerd en worden de requirements opgesteld. Hier is onder andere geïnterviewd welke sensoren er beschikbaar zijn om toe te passen. Er is gekozen (specifieke) firmware te maken voor 2 typen 3-draadssensoren voor het meten van luchtkwaliteit, omdat deze hiermee een grotere nauwkeurigheid hebben.

Om het systeem breder in te kunnen zetten worden er in fases die daarop volgen in 3 iteraties verbeteringen aangebracht aan de firmware van de GEN-node. Elke iteratie bestaat uit ontwerpen, ontwikkelen en testen.

Tot slot zijn de adviezen en conclusies opgesteld en onderbouwt.

## 2. Organisatie Technolution

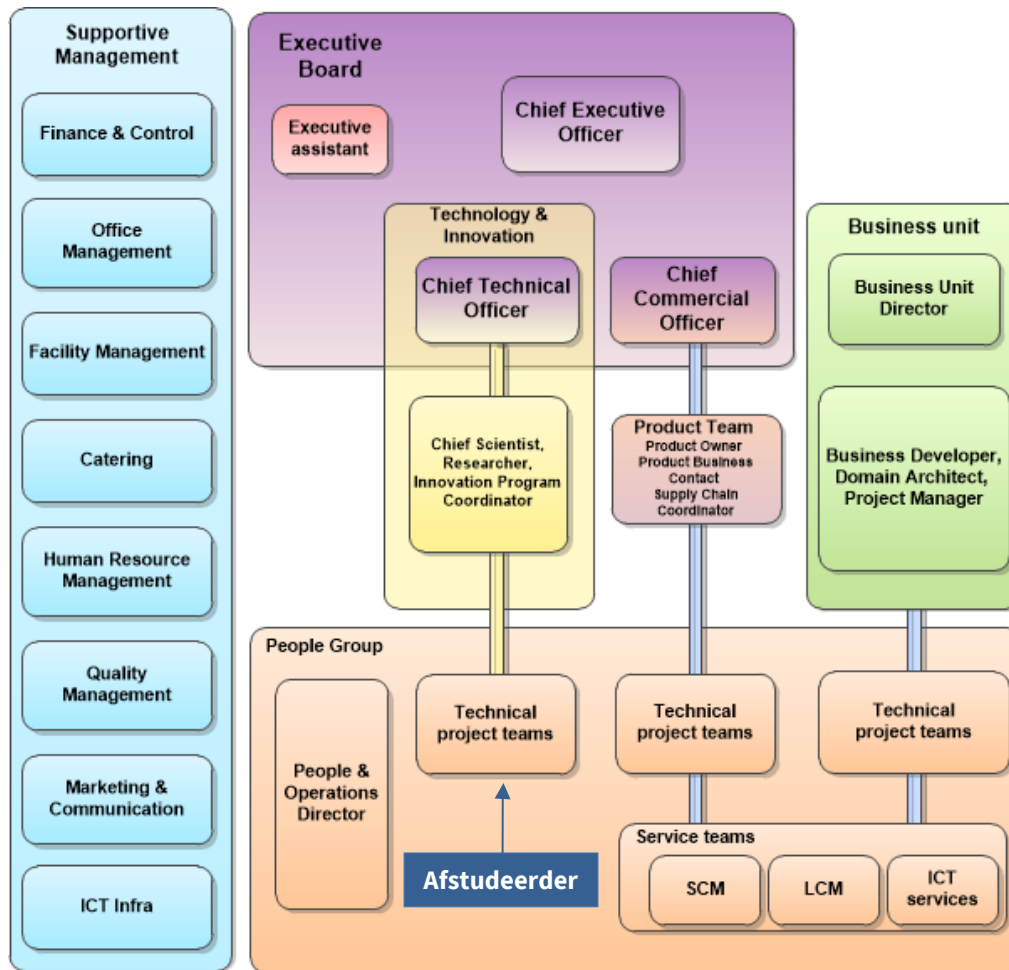
Technolution is een commercieel bedrijf dat zich richt op de ontwikkeling van elektronica, programmeerbare logica en (embedded) software voor diverse organisaties. De missie van het bedrijf is het maken van technische oplossingen die waardevol zijn voor de klant. De hoofdvestiging is in Gouda en er is tevens een vestiging in Deventer. Het afstuderen wordt gedaan in Gouda.

### 2.1 Organisatiestructuur

In de vestiging in Gouda werken meer dan 200 medewerkers. De projecten binnen het bedrijf zijn verdeeld over vier business units: mobiliteit, energie, industrie en Public Safety & Security. Dit project valt onder de business unit energie. Hier houdt het bedrijf zich bezig met het Internet of Things voor een slim energienetwerk. Al deze projecten zijn te vinden onder de 'technical project teams' in het organigram (afbeelding 2.1). Hier is ook mijn positie in de organisatie te vinden. De technical project teams worden aangestuurd door de hogere lagen in het organigram.

Binnen de organisatie heb ik te maken met de opdrachtgever Olaf Peters en begeleider Jordy Huiting. Olaf Peters heeft binnen het bedrijf de functie Domein Architect en Jordy Huiting is System Designer. De organisatiestructuur van Technolution is gebaseerd op de matrixorganisatie wat zich typeert door de indeling in business units en het projectgericht werken, waarbij elk project telkens opnieuw een eigen team en verantwoordelijken heeft. Binnen de projecten wordt een informele cultuur nagestreefd. Er zijn namelijk korte lijnen met direct betrokkenen en leidinggevende.

Alleen ikzelf werk aan de uitvoering van dit afstudeerproject. Ik word daarbij begeleid door Jordy Huiting. Olaf Peters geeft als opdrachtgever aan wat hij van dit project verwacht. Wekelijks heb ik met de opdrachtgever en begeleider een meeting. Met de begeleider zit ik tegenover hem in dezelfde kantoorruimte en ik hou op die manier regelmatig contact met hem. Ten tijde van de coronacrisis is dit contact digitaal voortgezet.



» Afbeelding 2.1, organigram Technolution (bron: Technolution Intranet)

## 2.2 Probleemdomein

Het probleemdomein van dit project is het overkoepelende project genaamd Sense2Grow. In het Sense2Grow-netwerk wordt door middel van sensoren informatie verzameld, opgeslagen in een database en weergegeven in een dashboard. Eén van de bedrijven die hier gebruik van maakt is Tata Steel, maar het wordt ook toegepast in kassen. De GEN-node is een onderdeel van dit netwerk. Aan Sense2Grow en de GEN-node hebben er naast de vaste medewerkers ook al een WO- en een HBO-stagiair aan gewerkt. Het laatste project was van de HBO-stagiair en was gericht op het verbeteren van de hardware van de GEN-node. Hierbij is een herontwerp gemaakt van de printplaat, met als resultaat ondersteuning van meerdere (analoge) sensoren. Software-ondersteuning van deze sensoren ontbreekt.

## 3. Opdracht

Dit hoofdstuk bestaat uit een beschrijving van de opdracht en de aanpak zoals deze is opgesteld in het afstudeerplan.

### 3.1 Aanleiding

Technolution heeft een bestaand IoT-netwerk Sense2Grow. In dit netwerk hangen LoRa-sensornodes, de GEN-nodes, die enkele sensoren ondersteunen. De ondersteuning van het aantal sensoren is momenteel beperkt en voor elk type sensor moet specifieke firmware worden geschreven. Technolution wil daarom een verbeterde GEN-node die tevens generieker is.

### 3.2 Probleemstelling

De hoofdvraag van dit onderzoek is:

Hoe kan de GEN-node verbeterd worden?

Om de hoofdvraag te beantwoorden dienen de volgende deelvragen te worden beantwoord:

- Hoe kunnen reeds geselecteerde sensoren worden ondersteund?
- Hoe kan de betrouwbaarheid verbeterd worden?
- Hoe kan de energiezuinigheid verbeterd worden?
- Hoe kunnen sensoren ten behoeve van het meten van luchtkwaliteit worden ondersteund?

### 3.3 Doelstelling

De afstudeeropdracht leidt tot verbeterde firmware voor een Proof-of-Concept-model GEN-node die energiezuiniger is, betrouwbaarder is en ondersteuning heeft voor eerder geselecteerde sensoren (voor de agrarische markt) plus geselecteerde sensoren voor het meten van luchtkwaliteit.

### 3.4 Resultaat

Het resultaat is een Proof-of-Concept-model van de derde generatie GEN-node waarbij het belang ligt bij verbeterde firmware.

### 3.5 Randvoorwaarden en afbakening

- De firmware is gebaseerd op de laatste versie van de software, geschreven in C.



- De broncode volgt het interne handboek C style guide en de software is opgebouwd in aparte modules.
- Het project richt zich op de software die op de GEN-node draait. De afstudeerder werkt met de aangeboden hardware, namelijk de laatste hardwareversie van de GEN-node.
- In dit project wordt er binnen het Sense2Grow-netwerk voornamelijk gefocust op de GEN-node.
- Elektrotechniek valt buiten de scope van de studie. Voor elektrotechnische vraagstukken, zoals het doen van energiemetingen, kan de hulp van de begeleider of een medewerker gevraagd worden.
- Het project heeft een duur van 17 weken.

### 3.6 Aanpak

Voor het uitvoeren van de afstudeeropdracht zijn er 70 dagen beschikbaar. Over de gehele periode van 85 dagen zijn 15 dagen, verdeeld over de periode beschikbaar voor het schrijven van het afstudeerverslag.

De volgende werkzaamheden worden tijdens dit project uitgevoerd waarbij de ontwerpfase, ontwikkelfase en de testfase een iteratief karakter zullen hebben. Het gaat hier om vier iteraties van de volgende onderwerpen: reeds geselecteerde sensoren, betrouwbaarheid, energiezuinigheid en te selecteren luchtkwaliteitssensoren.

#### **Uitvoeren afstudeeropdracht (70 dagen)**

- Oriëntatiefase – Product: Plan van Aanpak
  - Verhelderen opdracht
  - Risico's analyseren
  - Planning
- Analysefase – Product: Analyserapport
  - Literatuur bestuderen
  - Analyse huidige situatie
  - Requirements achterhalen
  - Systeemeisen vaststellen
- Ontwerpfase – Product: Ontwerprapport
  - Ontwerpen architectuur
- Ontwikkelfase – Product: Firmware Proof of Concept
  - Ontwikkelen van firmware.
- Testfase – Product: Testrapport
  - Uitvoeren van veldtesten.

- Adviesfase – Product: Adviesrapport
  - Conclusies en adviezen onderbouwen

**Schrijven afstudeerdossier (15 dagen, verdeeld over 85 dagen)**

- Afstudeerdossier (15 dagen)

## 4. Oriëntatiefase

Oriëntatie-fase	Analyse-fase	Iteratie 1	Iteratie 2	Iteratie 3	Advies-fase
-----------------	--------------	------------	------------	------------	-------------

In dit hoofdstuk wordt de oriëntatiefase beschreven. Deze fase heeft als resultaat de bijlage Plan van Aanpak. Tijdens deze fase wordt de opdracht verhelderd, de projectrisico's worden in kaart gebracht en wordt de planning gespecificeerd.

### 4.1 Verhelderen opdracht

De eerste stap is het helder krijgen van de opdracht. Ik wilde dit helder krijgen aan de hand van gesprekken met de opdrachtgever en begeleider. Tijdens de gesprekken heb ik gevraagd naar wat de GEN-node moet kunnen aan het einde van het project. Hieruit bleek dat de GEN-node ondersteuning moet krijgen voor in ieder geval analoge sensoren van het type 0 – 1 volt, 0 – 10 volt en 4 – 20 mA omdat dit sensoren zijn uit de industriestandaard. Dat was volgens de opdrachtgever een belangrijk onderdeel. Ook wilde de opdrachtgever dat er de mogelijkheid was om SPI- en I<sup>2</sup>C-communicatie te gebruiken, zodat hier ook sensoren voor toegepast kunnen worden. Deze sensoren zouden dan binnen 1 firmwareversie komen, door het toepassen van een mogelijkheid tussen het wisselen van sensoren zonder van firmware te veranderen. Analog Out- en Digital Out-ondersteuning moet blijven in de firmware.

Ook heb ik gevraagd naar de probleem- en doelstelling. Hieruit bleek dat de deelvraag over de reeds geselecteerde sensoren ging om een universele ondersteuning van analoge sensoren. Dat heeft er mee te maken dat de wens voor een generiekere GEN-node er al langer ligt. Daarnaast is er een klant van Technolution weggefallen waardoor noodzaak voor de ondersteuning van een aantal specifiek geselecteerde sensoren verviel.

Ik heb gevraagd wat er precies bedoeld wordt met betrouwbaarheid (of nauwkeurigheid). In het afstudeerplan was namelijk als tweede deelvraag de vraag afgestemd: Hoe kan de betrouwbaarheid verbeterd worden? Echter bleek zowel de opdrachtgever en de begeleider van mening dat de betrouwbaarheid geen verbetering behoeft. Het bleek een misverstand te zijn dat de betrouwbaarheid niet voldoende zou zijn. De deelvraag over de betrouwbaarheid kon daarom geschrapt worden volgens de opdrachtgever. Ik denk zelf dat er verwarring is ontstaan over het woord betrouwbaarheid omdat de betrouwbaarheid verbetert met het gebruikmaken van de nieuw geplaatste externe Analog/Digital Converter (ADC), in vergelijking met de interne ADC. Deze externe ADC zal gebruikt gaan worden voor de

uitwerking van de eerste deelvraag (universele ondersteuning analoge sensoren) en is daarom als aparte deelvraag overbodig geworden.

De deelvraag over geselecteerde sensoren ten behoeve van het meten van luchtkwaliteit bleef behouden. Luchtkwaliteit meten was nog steeds interessant voor de opdrachtgever. Tot slot zou er volgens de deelvraag over energiezuinigheid een onderzoek moeten komen naar hoe het energieverbruik verbeterd kan worden. Volgens de opdrachtgever en de begeleider bleek dat er het vermoeden was dat de software energiezuiniger geschreven kon worden. Omdat dit minder hoge prioriteit heeft dan de andere deelvragen is deze deelvraag als laatst geplaatst.

### **Nieuwe probleem- en doelstelling**

Aan de hand van deze antwoorden is de opdracht helderder geworden. De probleemstelling en het doel moest wel aangepast worden. De probleemstelling ziet er nu als volgt uit:

De hoofdvraag van dit onderzoek is:

Hoe kan de firmware van de GEN-node verbeterd worden?

Om de hoofdvraag te beantwoorden dienen de volgende deelvragen te worden beantwoord:

- Hoe kunnen analoge sensoren universeel worden ondersteund met sensorkeuze?
- Hoe kunnen geselecteerde sensoren ten behoeve van het meten van luchtkwaliteit worden ondersteund?
- Hoe kan het energieverbruik worden verlaagd?

En de doelstelling luidt: “De afstudeeropdracht leidt tot verbeterde firmware voor een Proof-of-Concept-model GEN-node die ondersteuning heeft voor analoge sensoren, plus geselecteerde sensoren voor het meten van luchtkwaliteit waarbij het energieverbruik verlaagd is.”

## **4.2 Risico's analyseren**

Om problemen tijdens het project te voorkomen of in te perken heb ik een risicoanalyse uitgevoerd. Hierbij zijn 5 risico's naar voren gekomen, deze zijn opgenomen in tabel 4.1 tot en met 4.5. Kans en impact staan aangegeven met sterren waarbij vijf sterren staat voor groot en één ster staat voor klein.

Risico	Opdrachtgever heeft gaandeweg het project een ander doel voor ogen
Kans	★★☆☆☆
Kansverlagende maatregel	Doelen goed vastleggen aan het begin van het project
Impact	★★★★★
Impactverlagende maatregel	Het nieuwe doel proberen in te passen in de requirements
Plan B	Ontwerpen aanpassen en vervolgfases opnieuw doorlopen

» Tabel 4.1

Risico	Het doel blijkt gaandeweg technisch niet haalbaar te zijn
Kans	★★☆☆☆
Kansverlagende maatregel	Vroegtijdig technisch onderzoek doen
Impact	★★★★★
Impactverlagende maatregel	Overleg om requirements te veranderen waar dit voor belemmering zorgt
Plan B	Het doel aanpassen

» Tabel 4.2

Risico	De werkzaamheden blijken complexer dan gepland
Kans	★★★★☆
Kansverlagende maatregel	Vroegtijdig technisch onderzoek doen
Impact	★★★★☆
Impactverlagende maatregel	Enkele requirements naar beneden schalen in prioriteit
Plan B	De complexe werkzaamheden veranderen in minder complexe werkzaamheden en de complexe werkzaamheden opnemen in het advies voor een vervolgproject

» Tabel 4.3

Risico	De GEN-nodes raken defect waardoor er niet getest kan worden
Kans	★★★★☆
Kansverlagende maatregel	Back-up hardware hebben
Impact	★★★★★

Impactverlagende maatregel	Oudere hardware gebruiken
Plan B	Zonder hardware verder werken aan ontwerpen totdat deze weer gereed is

» Tabel 4.4

Risico	Er is onvoldoende geheugen op de hardware
Kans	★★★★☆
Kansverlagende maatregel	Rekening houden bij het ontwerp met de beperkte geheugenruimte
Impact	★★★★☆
Impactverlagende maatregel	Minder gebruikte softwaredelen schrappen uit de firmware
Plan B	Adviseren andere hardware te gebruiken of de firmware op te delen voor verschillende functionaliteiten

» Tabel 4.5

## 4.3 Planning

Bij het maken van een planning is het nodig om te weten welke softwareontwikkelingsmethode er tijdens het project gebruikt gaat worden. Ik heb gekozen voor Rapid Application Development (RAD). Vervolgens wordt de deze methode toegepast op de fasering.

### 4.3.1 Methode

Ik heb de keuze van de methode die past bij dit project af laten hangen van de ontwikkelstrategie.

Er zijn vier ontwikkelstrategieën: acceptatie, lineair, iteratief, incrementeel. Dit project bestaat uit drie onderdelen die apart van elkaar ontworpen, ontwikkeld en getest kunnen worden. Daarom past dit project binnen een iteratieve en een incrementele strategie. Een incrementele ontwikkelstrategie is geschikt bij een project waarbij de ontwikkelaar onbekend is met de gebruikte technologie. Bij een iteratieve strategie zijn de doelstellingen vaag, wat niet het geval is in dit project. Daarom is er gekozen voor een incrementele ontwikkelstrategie.

Bij de incrementele ontwikkelstrategie passen enkele methodes: RAD, RUP, Open UP, Scrum, DSDM en XP. Ik heb deze methodes individueel beoordeeld in tabel 4.6. Veel van deze methodes zijn vooral geschikt voor grootschalige projecten. Wegens de beperkte grootte van het project (1 ontwikkelaar, 1 opdrachtgever) en de beperkte doorlooptijd (17 weken) is RAD

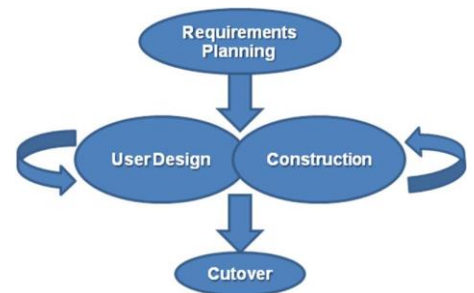
erg geschikt aangezien de focus ligt bij het snel ontwikkelen in een kleinschalig project. Daarom heb ik gekozen voor RAD.

Methode	RAD	RUP	Open UP	Scrum	DSDM	XP
Projectgrootte	+	0	+	-	0	-
Doorlooptijd	+	0	+	0	0	+
Test per iteratie	+	+	+	-	+	+
Modulair programmeren	+	0	0	0	0	-
Afgebakend doel	+	+	0	-	-	-

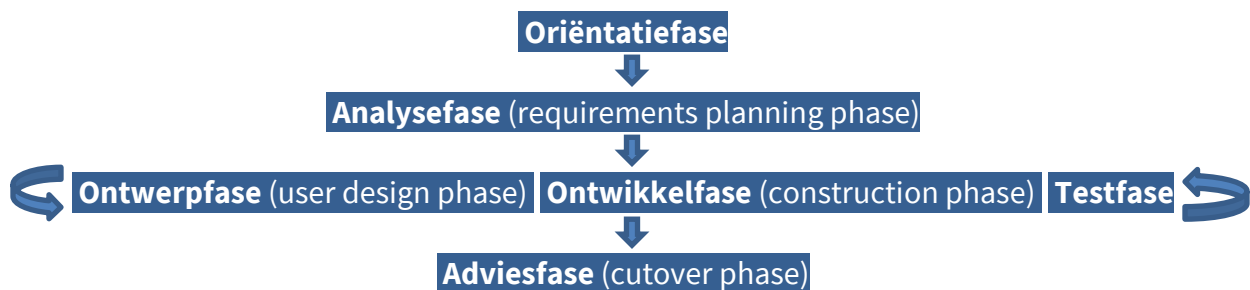
» Tabel 4.6, Beoordeling per methode. +: goed, 0: niet optimaal, -: onvoldoende

### 4.3.2 Fasering

De vier fases die bij RAD horen zijn opgenomen in afbeelding 4.1. Voor dit project heb ik ervoor gekozen om RAD aan te passen zodat RAD beter aansluit bij dit project. Er is namelijk sprake van ontbrekende kennis die in de analysefase wordt opgedaan. Ook sluit de methode dan beter aan bij de werkzaamheden die afgesproken zijn in het afstudeerplan. Er zijn ook geen gebruikers waardoor de suggesties en wensen alleen van de opdrachtgever komt in de fases. De testfase is uit de construction phase gehaald, zoals dit in het afstudeerplan ook is vermeld. De fasering met de aangepaste versie van RAD is zichtbaar in afbeelding 4.2.



» Afbeelding 4.1, fasering RAD (bron: Wikimedia.org, File:RADModel.jpg)



» Afbeelding 4.2, aangepaste fasering RAD

De planning voor dit project is opgenomen in tabel 4.7. Dit project is uiteindelijk ingedeeld in 3 iteraties. Bij elke iteratie wordt er antwoord gegeven op een deelvraag:

- Iteratie 1: Hoe kunnen analoge sensoren worden ondersteund met sensorkeuze?

- Iteratie 2: Hoe kunnen sensoren ten behoeve van het meten van luchtkwaliteit worden ondersteund?
- Iteratie 3: Hoe kan er rekening worden gehouden met energiezuinigheid?

Week	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Oriëntatiefase																	
Analysefase																	
Iteratie 1																	
Iteratie 2																	
Iteratie 3																	
Adviesfase																	

» Tabel 4.7, Planning

De analysefase behandelt ook de analyses die nodig zijn voor de drie iteraties. Ik heb hiervoor gekozen omdat er tussen de iteraties een grote samenhang is. Het is daarom wenselijk om alvast enige kennis te hebben van de drie vraagstukken. Tijdens elke iteratie wordt de ontwerpfase, ontwikkelfase en de testfase doorlopen. Aan het einde van elke fase wordt bijbehorend tussenproduct opgeleverd. Deze staan vermeld in hoofdstuk 3.6 Aanpak.



## 5. Analysefase

Oriëntatie-fase	Analyse-fase	Iteratie 1	Iteratie 2	Iteratie 3	Advies-fase
-----------------	--------------	------------	------------	------------	-------------

Voor de analysefase doe ik onderzoek naar de huidige GEN-node. Zodra de huidige situatie duidelijk is kan ik gericht analyseren wat nodig is voor de gewenste GEN-node. Met deze informatie ga ik de requirements achterhalen en worden de systeemeisen vastgesteld. Dit heeft als resultaat de bijlage Analyserapport.

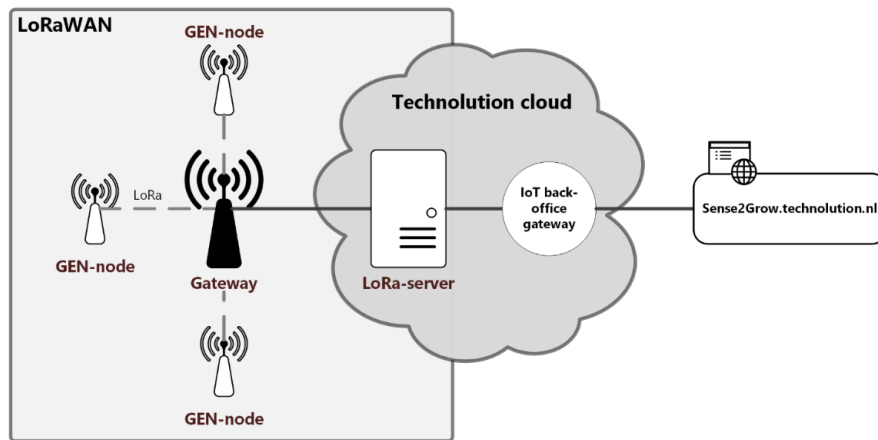
### 5.1 Huidige GEN-node

Omdat de GEN-node voor mij voor aanvang van het project nog een vrij onbekend begrip was, wil ik graag te weten komen wat de context is waarin de GEN-node gebruikt wordt en hoe de GEN-node is opgebouwd. Hiermee krijg ik een duidelijker beeld van de huidige situatie. Aangezien er over de GEN-node geschreven is in interne documentatie en datasheets, zal het onderzoek plaats vinden aan de hand van literatuurstudie.

De hardware van de tweede generatie GEN-node is het resultaat van een eerdere stageopdracht [1]. De hardware die hierbij gerealiseerd werd zal worden gebruikt tijdens dit project. Deze generatie van de sensornode heeft ten opzichte van de oudere variant aanpassingen die voorsorteren op een GEN-node die een grotere ondersteuning heeft voor analoge sensoren en de mogelijkheid om instellingen te selecteren op de hardware.

#### 5.1.1 Sense2Grow-netwerk

De GEN-node is onderdeel van het Sense2Grow-netwerk. Dit is een netwerk dat oorspronkelijk bedoeld is voor het gebruik in de agrarische sector. Hiermee kan bijvoorbeeld de temperatuur verspreid door een kas worden gemeten. Eén van de ondersteunde sensoren wordt dan aangesloten op de GEN-node, zoals een temperatuursensor. De GEN-node leest de sensor elke vijf minuten uit en stuurt zijn gegevens door, zodat de gegevens uitgelezen kunnen worden in het realtime dashboard op [sense2grow.technolution.nl](https://sense2grow.technolution.nl). Een overzicht van dit netwerk is te vinden in afbeelding 5.1.

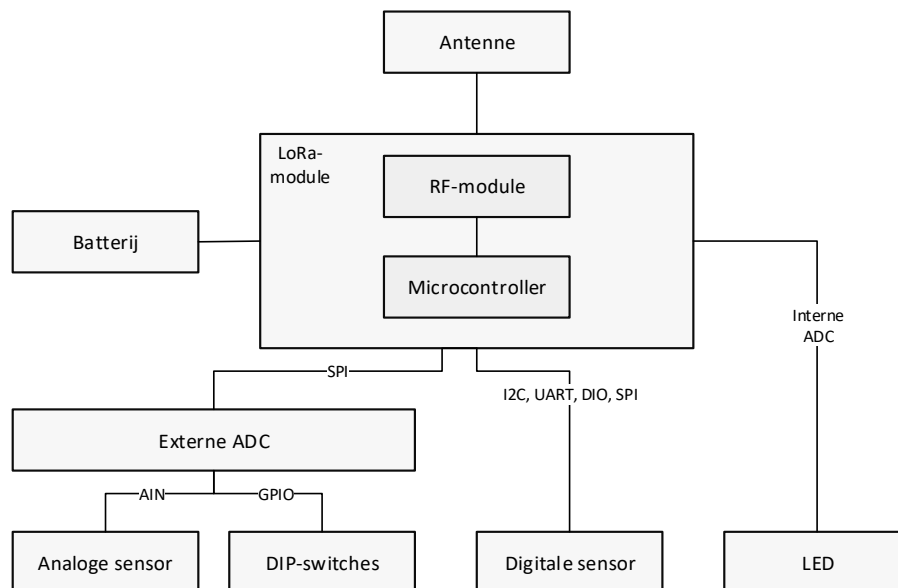


» Afbeelding 5.1, Sens2grow-netwerk

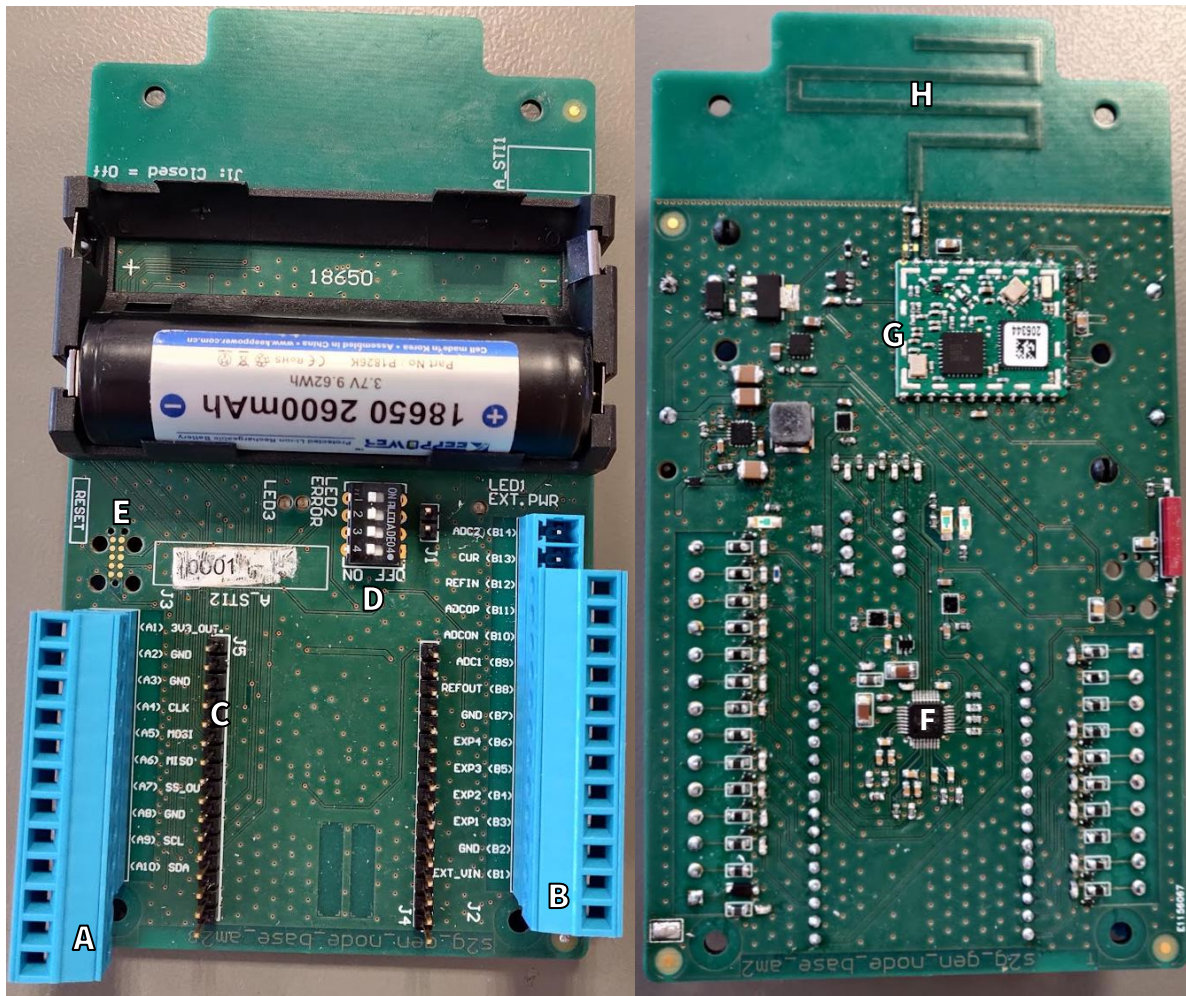
De draadloze communicatie vanuit de node is gebaseerd op het LoRa-protocol. Elke node maakt verbinding met een LoRa-gateway. Voor het netwerk is er minstens 1 gateway nodig die bij de klant geïnstalleerd wordt. De gateway stuurt de gegevens door over het internet naar de LoRa-server. Deze server stuurt de gegevens door naar de backoffice waar ze bewaard worden.

### 5.1.2 Printed Circuit Board (PCB)

De GEN-node is een PCB (printplaat) waarop modules en sensoren aangesloten kunnen worden. De PCB is een ontwerp van Technolution. Ik heb de interne documentatie en de datasheet van de LoRa-module [2] bestudeerd. Vervolgens heb ik hiermee een overzicht gemaakt van de onderdelen op de PCB, zie afbeelding 5.2. In afbeelding 5.3 is de voor- en achterzijde te zien van de GEN-node met daaronder een legenda.



» Afbeelding 5.2, hardware-overzicht PCB



» Afbeelding 5.3, boven- en onderaanzicht PCB

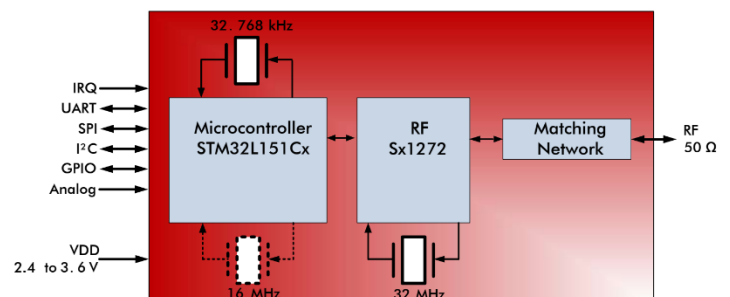
- |                                      |                                    |
|--------------------------------------|------------------------------------|
| <b>A.</b> Schroefconnectoren A-zijde | <b>E.</b> Programmeerheader        |
| <b>B.</b> Schroefconnectoren B-zijde | <b>F.</b> ADS114S06B ADC           |
| <b>C.</b> Expansieheader             | <b>G.</b> WiMOD iM880B LoRa-module |
| <b>D.</b> DIP-switches               | <b>H.</b> Antenne                  |

Legenda afbeelding 2.2

De belangrijkste onderdelen op afbeelding 5.3 zijn de LoRa-module (G), de ADC (F) en de DIP-switches (D).

### LoRa-module

De LoRa-module (G) is een WiMOD iM880B [3]. Deze module bevat de microcontroller STM32L151C6 en de RF-transceiver (Sx1272) voor het gebruik van de LoRa-antenne [2], zie ook afbeelding 5.4. De



» Afbeelding 5.4, schematische tekening iM880B LoRa-module [2]

microcontroller bevat in het flashgeheugen de firmware voor de GEN-node die deels bestaat uit een statische bibliotheek (static library) die origineel bij de module hoort. De module bevat diverse interfaces zoals SPI en I<sup>2</sup>C voor digitale communicatie en GPIO.

### **Analog Digital Converter (ADC)**

De ADC wordt gebruikt voor het omzetten van analoge signalen naar digitale signalen met als doel de spanning te kunnen meten. De microcontroller bevat een interne ADC.

Nieuw toegevoegd is de externe ADC (F) op deze generatie GEN-node. Het gaat om de ADS114S06B van Texas Instruments en wordt aangestuurd via SPI [4]. Het heeft een Programmable Gain Amplifier (PGA). Hiermee kan de versterking aangestuurd worden via de SPI en kunnen dus ook kleine spanningen uit sensoren goed worden gemeten. Ook heeft deze 16-bits ADC heeft een resolutie van 32767 ( $2^{15} - 1$ ). Dit zorgt beiden voor een hogere nauwkeurigheid in tegenstelling tot de interne ADC waar geen PGA op aanwezig is en een lagere resolutie (4095). Voor de externe ADC is momenteel alleen nog eenvoudige testfirmware gemaakt om de hardware te kunnen testen.

Wanneer de ADC een spanningsmeting heeft gedaan geeft de ADC de waarde niet in volt terug, maar een binair getal. Om deze waarde om te kunnen zetten naar volt, zal in de nieuwe firmware gebruik gemaakt moeten worden van de volgende vergelijking [5]:

$$\frac{\text{Resolutie ADC}}{\text{Referentiespanning}} = \frac{\text{ADC-waarde}}{\text{Gemeten spanning}}$$

Als de interne referentiespanning wordt gebruikt op deze ADC (2,5 V) is de gemeten spanning dus te berekenen met deze formule:

$$\text{Gemeten spanning} = 2,5 \text{ V} * \frac{\text{ADC-waarde}}{32767}$$

### **DIP-switches**

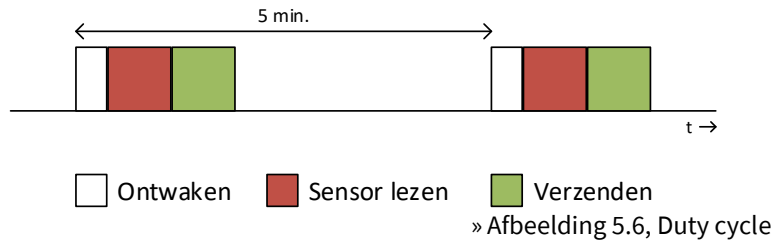
De PCB heeft een viertal DIP-switches (D). Dit zijn kleine, handmatige schakelaars met een aan- en een uitstand. Zo zijn er 16 mogelijkheden. De switches zijn op de PCB gezet met het doel een sensortype te kunnen selecteren. Momenteel worden de switches nog niet gebruikt in de firmware. De switches zijn aangesloten op de 4 GPIO-ingangen van de externe ADC.

Voor de overige modules en pin-layouts, zie hoofdstuk 2.2 PCB in de bijlage Analyserapport.

### **5.1.3 Software**

De huidige software is zodanig opgebouwd om 1 sensortype uit te lezen per firmwareversie. Bij het uploaden van de firmware naar de GEN-node wordt er gekozen voor welke sensor dit





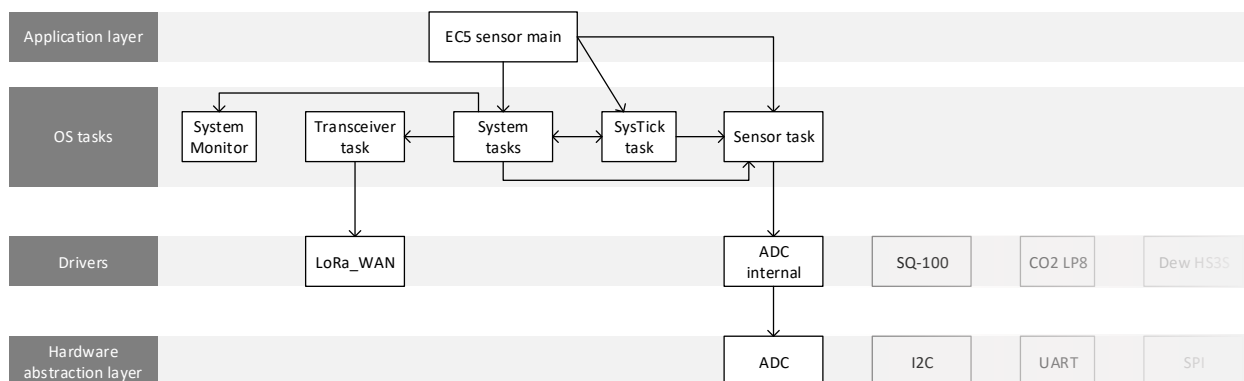
## Taken

In het systeem zijn er ook verschillende taken die aangeroepen worden.

- SysTick Task: Ontwaakt het apparaat en start andere taken
- Sensor Task: Regelt dat de sensorwaardes uitgelezen worden, maakt een bericht en plaatst in buffer
- System Monitor Task (SysMon): Monitort het apparaat en verzamelt apparaatstatistieken zoals batterijwaarde en interne temperatuur
- Transceiver Task: Verzend berichten in buffer met LoRa

## Architectuur

Ik heb de softwarearchitectuur bestudeerd. Hiervoor waren verder geen ontwerpen van beschikbaar bij Technolution. Om een eerste beeld te krijgen van de huidige software heb ik daarom een overzicht gemaakt van de softwarelagen, -elementen en -relaties, zie afbeelding 5.7. De afbeelding laat een vereenvoudigd totaaloverzicht van de software zien.



» Afbeelding 5.7, elementen en relaties EC5-firmware

Het element in de applicatielaag bevat een main-functie. Hiermee kan de software aangepast worden voor een bepaalde sensor. Bij het uploaden naar de node wordt er gekozen welke applicatie bij het apparaat hoort. Als er een andere sensor aangesloten wordt, moet er dus nieuwe firmware geüpload worden.

De onderlinge relaties in afbeelding 5.7 bestaan wanneer er firmware voor een sensor wordt toegepast. Als voorbeeld is er gekozen voor de EC5-sensorapplicatie. Hierdoor is er een specifieke main-functie en een specifieke sensortask afgestemd op deze sensor.



## **Terugkeren naar het OS**

Kenmerkend is dat de software zodanig is geschreven dat een taak (zoals een sensortask) steeds opnieuw aangeroepen wordt door het OS. Dit gaat door totdat de taak het signaal geeft dat de taak voltooid is. Dit wordt ook wel het herhaaldelijk terugkeren naar het OS genoemd. De taken bestaan daarom uit 1 of meerdere state machines. De reden dat dit zo geschreven is heeft te maken met de watchdog. De watchdog heeft als doel te detecteren dat software blijft hangen, om zonodig een reset uit te voeren. Door terug te keren naar het OS weet de watchdog dat het systeem nog steeds draait.

## **5.2 Onderzoek gewenste firmware GEN-node**

Vervolgens heb ik onderzoek uitgevoerd gericht op de deelvragen van dit project. Hierbij wordt de informatie vergaard die nodig is voor het uitvoeren van de vervolgfases. Dit zorgt er voor dat er een beter beeld is van de mogelijkheden en mogelijke obstakels van het geheel. En dat komt van pas bij het achterhalen van de requirements in hoofdstuk 5.3 en eventueel aanvullend onderzoek in de ontwerpfases. Ik heb net als in de vorige paragraaf literatuurstudie gebruikt, maar dit keer gericht op de deelvragen van de probleemstelling.

### **5.2.1 Analoge sensoren universeel ondersteunen**

Voor de nieuwe firmware is het een requirement om analoge sensoren te ondersteunen zonder dat er voor iedere sensor een specifiek stuk software moet worden geschreven. Een generieke ondersteuning voor een groot aantal analoge sensoren is gewenst. Dit zal moeten gebeuren op de nieuwe ADC. Een analoge sensor toont zijn resultaat door de stroom of de spanning van het uitgangssignaal te veranderen. De schaal waarbinnen de minimale en maximale waardes vallen, verschillen per type sensor. Veelvoorkomende analoge uitgangssignalen van industriële sensoren zijn 4 tot 20 mA, 0 tot 5 V en 0 tot 10 V.

Omdat we te maken hebben met verschillende meetmethodes per type analoge sensor, moet er op de PCB een keuze kunnen worden gemaakt hoe er uitgelezen moet worden. In een eerder project is er gekozen om dit te kunnen doen door middel van DIP-switches. Deze maken het mogelijk te schakelen tussen 16 voorinstellingen.

### **Analoge sensoren en voorinstellingen**

Om inzicht te krijgen in relevante sensoren die voor kunnen komen heb ik een lijst opgesteld, zie tabel 5.1. Samen met mijn begeleider is de lijst compleet gemaakt. Deze lijst bestaat uit sensoren die gebruikt zijn bij klanten, waarvoor interesse bij klanten is of waarvoor onderzoek is gedaan in eerdere projecten. Voor deze onderzoeksvraag is het witte gedeelte van de tabel (5.1) met analoge sensoren van toepassing. Het lichtgrijze gedeelte zijn sensoren die al ondersteuning hebben (komen terug in hoofdstuk 6.1.2). En onderin in de tabel in het

donkergrijze vlak, samen met de driedraadsvariant van de PT500, staan sensoren die in hoofdstuk 7 aan bod komen.

Voor de maximale uitgangsspanningen van 2.5V en lager kan er gekozen worden voor de interne referentiespanning van 2,5 V op de ADC. Hierbij is voor sommige sensoren een versterking nodig (gain). Met de externe ADC is een versterking tot 128x mogelijk [4]. Voor de sensoren met een maximale uitgangsspanning van 3,3V en 5V kan er een externe referentiespanning aangesloten worden. Er zijn ook sensoren met een maximale spanning tot 10 V, maar ik zag dat dit een probleem oplevert. De analoge ingangen van de ADC kunnen een spanning aan van maximaal 5,8 V [4]. Dit betekent dat de uitgangsspanning van de sensor naar beneden gebracht moet worden. De hoge spanningen kunnen teruggeschaald worden naar 2,5 V of 5 V. Voor die laatste optie heb ik niet gekozen aangezien er bij een maximale uitgangsspanning van 5V een externe referentiespanning vereist is. Dit is minder gebruiksvriendelijk.

Sensor	Functie	Communicatie	Uitgang	R ( $\Omega$ )	V <sub>resolutie</sub>	Gain
PT500	Temperatuur	Analoog	460-620 $\Omega$	-	0,31 V	8
EC5	Bodemvocht	Analoog	V <sub>cc</sub> (10% - 50%) V	-	1,25 V	2
SQ-500	PAR (licht)	Analoog	0 – 40 mV	-	0,040 V	32
Licor LI-190	PAR	Analoog	0-0.050mA – 0-0.100mA	604	0,0604 V	32
TZW-330 18-03	Luchtdrukverschil	Analoog	4 – 20 mA	100	2 V	1
SMP6-a	Pyranometer	Analoog	4 – 20 mA	100	2 V	1
-	Dummy	Analoog	0 – 10 V	-	2,5 V	1
Ds18b20	Temperatuur	1-wire	-			
Sh31	Temperatuur	I2C	-			
LP8	CO <sub>2</sub>	I2C	-			
Hs3s	Dauw	I2C	-			
SPS30	PM	I2C	-			
RE46C190	Rook	GPIO	-			
NO2-B43F	NO <sub>2</sub>	Analoog	0 - 5 V (2x)			
SO2-B4	SO <sub>2</sub>	Analoog	0 - 5 V (2x)			

» Tabel 5.1, Lijst van sensoren

Voor sensoren die variëren in stroom geldt een andere aanpak. Hiervoor moet een bekende weerstand worden aangesloten in de stroomkring, zodat de ADC de spanning over de weerstand kan meten. Door de stroom te vermenigvuldigen met de weerstandswaarde wordt inzichtelijk wat de ingangsspanning zal worden. Voor de 4 – 20 mA sensoren zou dit met de ingebouwde weerstand op de PCB van 100 ohm betekenen dat de maximale uitgangsspanning 2 V is. Hiervoor is geen gain nodig. Een weerstand van 120 ohm geeft een



uitgangsspanning van 2,4 V, wat voor een grotere precisie zou zorgen, maar dan moet de ingebouwde weerstand vervangen worden of 20 ohm worden toegevoegd. Voor zwakkere signalen is een grotere gain nodig en/of een grotere weerstandswaarde.

Het lijkt dus nodig dat er met de DIP-switches gekozen kan worden een bepaalde gain in te stellen. Aan de hand van de te gebruiken sensoren, de aanbevolen weerstanden en de spanningsresolutie is de gain bepaald in tabel 5.1. Een gain van 1 (geen), 2, 8 en 32 komen voor in deze tabel. Deze zijn in ieder geval nodig om op de DIP-switches instelbaar te maken. Hoe beter de gain past bij de spanningsresolutie, hoe preciezer het resultaat. Afhankelijk van de wensen van de opdrachtgever en de vrije plekken op de DIP-switch kunnen er meer gain-instellingen worden gebruikt. De DIP-switches bieden 16 mogelijkheden.

### Hardware-aanpassingen

In overleg met mijn begeleider hebben we ervoor gekozen om 10 volts en 5 volts spanningen naar beneden te gaan brengen middels een hardware-aanpassing. Omdat zo'n hardware-aanpassing tijd kost is dit direct proces direct gestart. Mijn begeleider heeft een hardwarecollega de opdracht gegeven voor deze aanpassing. Ze kozen voor extra DIP-switches waarmee in dit geval hardwarematig tussen schakelingen kan worden gewisseld om de spanning onder of gelijk aan 2,5V te houden.

### 5.2.2 Geselecteerde luchtkwaliteitssensoren ondersteunen

Technolution is in het bezit van twee sensoren die gewenst zijn voor het meten van luchtkwaliteit in het Sense2Grow-netwerk. Het gaat om een sensor die stikstofdioxide meet ( $\text{NO}_2$ ) en een sensor die zwaveldioxide meet ( $\text{SO}_2$ ). De types zijn NO2-B43F en



» Afbeelding 5.8, NO2-B43F en SO2-B4

SO2-B4. Dit zijn de laatste twee sensoren uit de tabel 5.1, zie ook tabel 5.2 voor specificaties. Beide sensoren zijn analoge sensoren. Hierdoor zouden ze aangesloten kunnen worden als gewone analoge sensor. Het verschil is dat deze sensoren niet één maar twee uitgangssignalen heeft. Samen met het ingangssignaal van de sensor levert dit een driedraads sensor op. Beide sensoren zijn gemonteerd op een Alphasense B4 Individual Sensor Board (ISB) [7].

Een driedraadssensor levert een nauwkeurigere weergave omdat hiermee onder andere temperatuursinvloeden op de draden kan worden weggefilterd. Op het eerste draad wordt de voeding aangesloten. Op het tweede draad wordt de draadsweerstand gemeten. Op het derde draad wordt de meting en de draadsweerstand gemeten. Wordt de draadsweerstand van de meetwaarde van het derde draad afgetrokken, dan is het resultaat een meting zonder de invloeden van de draadsweerstand. [8]

	NO <sub>2</sub> -B43F [9]	SO <sub>2</sub> -B4 [10]
<b>Functie</b>	NO <sub>2</sub>	SO <sub>2</sub>
<b>Communicatie</b>	Analoog	
<b>Uitgangssignaal</b>	0 – 5 V	
<b>Referentiespanning</b>	Intern (2,5 V)	
<b>Benodigde gain</b>	1	

» Tabel 5.2, Specificaties geselecteerde sensoren

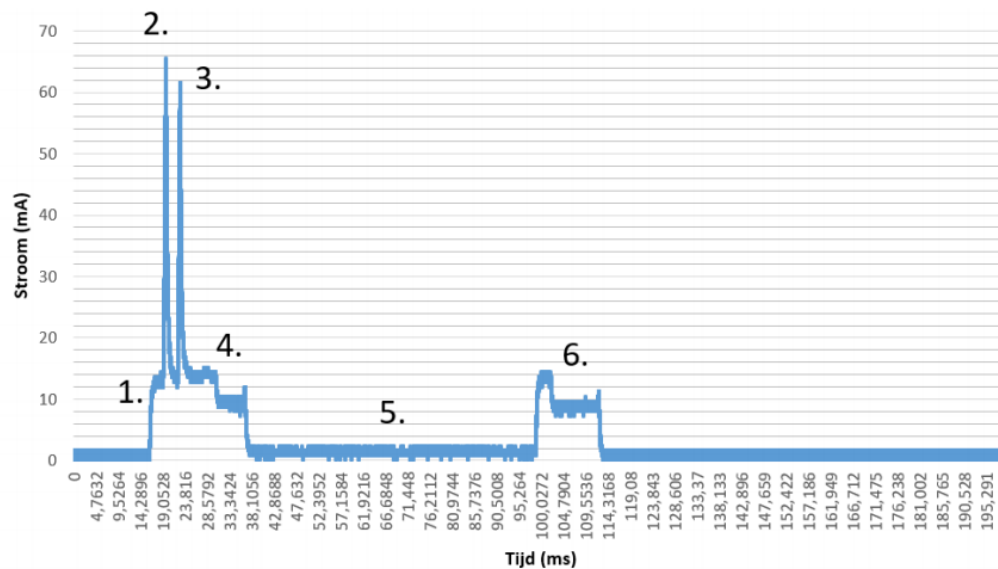
### 5.2.3 Energieverbruik verbeteren

Momenteel is de testsoftware die geschreven is voor de ADC niet optimaal wat betreft het energieverbruik volgens de opdrachtgever. Naast het punt dat het terugkeren naar het OS tijdens het uitvoeren van de taak veel energie zou kosten, zou ook het lezen van een analoge sensor sneller moeten kunnen. Dit zou ook energie moeten besparen.

Op hardwaregebied zijn er op verschillende vlakken al ontwerpkeuzes gemaakt die het energieverbruik ten goede komen. Zo worden verschillende onderdelen spanningsloos gemaakt wanneer de node in slaap valt. Dit is de volledige ADC en de referentiespanning [1].

Het energieverbruik bij een aantal sensoren was al in kaart gebracht [1]. Zo is het energieverbruik bij het gebruik van een sensor op de externe ADC is te zien in afbeelding 5.9. De oppervlakte (stroom × tijd) van deze grafiek is 0,00016 mAh. Hierin zijn de volgende gebeurtenissen te zien:

1. Systeem ontwaakt.
2. De spanning over de ADC wordt ingeschakeld. Dit veroorzaakt een stroompiek door het laden van de condensatoren.
3. De referentiespanning wordt ingeschakeld. Hier is ook weer een piek te zien door de condensator.
4. De microcontroller stuurt een START commando naar de ADC en gaat terug naar slaapmodus
5. ADC-conversie, ADC maakt een sample en zet dit om naar 16-bits.
6. LoRa-module stuurt commando om de samplewaarde uit te lezen. De waarde wordt uitgelezen door de LoRa-module.



» Afbeelding 5.9, stroom tijdens het uitlezen van een analoge sensor [1]

In een voorgaand project zijn er twee aanbevelingen gedaan om het energieverbruik te verbeteren. De eerste is de tijd beperken waarin de microcontroller actief is. In afbeelding 5.9 is te zien dat de het systeem er lang over doet om een sensorwaarde te meten (4, 5, 6).

Volgens de opdrachtgever en begeleider kost het terugkeren naar het OS tijdens het uitvoeren van een taak onnodig veel microcontrollertijd. Om de microcontroller minder lang actief te houden kan er indien mogelijk ook voor worden gekozen om de SPI-klok sneller te laten lopen en om bepaalde acties in te korten of om die niet steeds te hoeven initialiseren.

De tweede aanbeveling is het verlagen van de zogeheten oversampling ratio op de ADC. Hoe lager deze ratio, hoe hoger de datarate en dus ook de conversietijd. Bijkomend effect is wel, omdat er minder samples worden genomen, dat de nauwkeurigheid afneemt omdat de ruis toeneemt. [4]

Een kleine voedingsstroom van bijvoorbeeld 0,5 mA voor sensoren zoals de PT500 helpt ook bij het laag houden van het energieverbruik. De gain-mogelijkheden op de ADC maakt dat dit mogelijk is zonder grote verliezen in nauwkeurigheid.

Tot slot zou wellicht ook het proces van meten en versturen veranderd kunnen worden. Er zouden dan eerst metingen opgespaard kunnen worden om vervolgens met een kleinere frequentie een hoeveelheid metingen in 1 keer op te sturen naar de backoffice.

In hoofdstuk 8 worden deze onderwerpen verder onderzocht en getest door middel van stroommetingen.

## 5.3 Requirements achterhalen

Samen met de opdrachtgever heb ik besproken waaraan de GEN-node moet voldoen aan het einde van het project. Hierbij kwamen 3 punten sterk naar voren:

- Industriestandaard analoge sensoren moeten ondersteund worden.
- Ook andere en al gebruikte analoge sensoren worden voor zoveel mogelijk ondersteund.
- Er is een sterk vermoeden dat de software veel processortijd kwijt is door het steeds terugkeren naar het OS tijdens het uitvoeren van een taak, wat ten koste gaat van het energieverbruik. Grote stukken software zou herschreven moeten worden.

Er is samen met de begeleider invulling gegeven aan wat de te selecteren sensoren worden die een specifiekere ondersteuning nodig hebben. Hierbij was de voorkeur gevallen vanuit de begeleider op de NO<sub>2</sub>- en SO<sub>2</sub>-sensoren uit hoofdstuk 5.2.2. Deze waren namelijk al eerder aangeschaft om luchtkwaliteit in kaart te kunnen brengen.

Vervolgens is hier een requirementsvoorstel uit gekomen. De feedback, namelijk dat er requirements mistten over de kwaliteit van de firmware zelf, is meegenomen en de requirements zijn opnieuw voorgelegd. Dit heeft uiteindelijk geleid tot de functionele requirements die te vinden zijn in tabel 5.3 en niet-functionele requirements in de tabel 5.4. De requirements zijn geprioriteerd volgens de MoSCoW-methode, waarin een 'must have' de hoogste prioriteit heeft.

### 5.3.1 Functionele requirements

ID	Requirement	Prioriteit
R01	De firmware heeft driverondersteuning voor de ingangen Analog In, SPI en I2C en de uitgangen Analog Out en Digital Out.	Must
R02	De firmware heeft een implementatie voor het aansturen en het uitlezen van de externe ADC.	Must
R03	De firmware geeft de keuze voor het gebruik van analoge sensoren, zodat er ondersteuning is voor een sensoren met een uitgangssignaal van 4-20 mA, 0-2,5 V, 0-5 V en 0-10 V.	Must
R04	De firmware geeft de keuze voor het toepassen van de benodigde gain, zodat er ondersteuning is voor analoge sensoren met een zwak uitgangssignaal.	Must
R05	De firmware heeft ondersteuning voor geselecteerde driedraadssensoren (NO <sub>2</sub> en SO <sub>2</sub> ).	Must
R06	Uit de ondersteunde ingangssignalen van sensoren kan worden gekozen	Should

	zonder dat hiervoor gewisseld hoeft te worden van firmware.	
<b>R07</b>	De firmware heeft de keuze een kleine uitgangsstroom in te schakelen van 0,5mA waarmee de PT500-sensor gevoed kan worden.	Should
<b>R08</b>	De firmware heeft generieke ondersteuning voor SPI- en I <sup>2</sup> C-sensoren.	Could

» Tabel 5.3, Functionele requirements

### 5.3.2 Niet-functionele requirements

ID	Requirement	
<b>R09</b>	Het energieverbruik is lager dan het huidig verbruik wanneer de GEN-node wordt toegepast voor het uitlezen van analoge sensoren met de externe ADC.	Must
<b>R10</b>	De firmware is modulair opgebouwd zodat de modules herbruikbaar zijn.	Should
<b>R11</b>	De firmware is uitbreidbaar zodat er in de toekomst ook sensoren met andere communicatieprotocollen toegepast kunnen worden.	Should
<b>R12</b>	De firmware is zodanig ontworpen dat het flexibiliteit biedt voor toekomstig gebruikte analoge sensoren	Should

» Tabel 5.4, Niet-functionele requirements

## 5.4 Systeemeisen vaststellen

Als laatst heb ik de systeemeisen vastgesteld. Omdat de firmware moet draaien op de hardware van de GEN-node zijn de systeemeisen afgestemd op deze node. De derde generatie is de meest recente node. De GEN-node bevat een microcontroller die een beperkt flashgeheugen en RAM-geheugen heeft. Daarom is het van belang dat de firmware aan de limieten van de hardware voldoet.

- GEN-node generatie 3 of nieuwer
- 128 kB flash
- 32 kB RAM

## 6. Iteratie 1: ondersteunen analoge sensoren

In de eerste iteratie zal de focus komen te liggen op het ondersteunen van veelvoorkomende analoge sensoren op een universele manier.

### 6.1 Ontwerpfase

Oriëntatie-fase	Analyse-fase	Iteratie 1	Iteratie 2	Iteratie 3	Advies-fase
		Ontwerpfase	Ontwikkelfase	Testfase	

Het ontwerp in deze en komende ontwerpfases bouwen voort op bestaande software en bijbehorende ontwerpen. De bestaande software is in hoofdstuk 5.1.3 geanalyseerd. De ontwerpen zijn ook terug te vinden in de bijlage Ontwerprapport.

Bij deze iteratie wordt er vooral de focus gelegd op het realiseren van de onderstaande requirements.

<b>R02</b>	De firmware heeft een implementatie voor het aansturen en het uitlezen van de externe ADC.
<b>R03</b>	De firmware geeft de keuze voor het gebruik van analoge sensoren, zodat er ondersteuning is voor een sensoren met een uitgangssignaal van 4-20 mA, 0-2,5 V, 0-5 V en 0-10 V.
<b>R04</b>	De firmware geeft de keuze voor het toepassen van de benodigde gain, zodat er ondersteuning is voor analoge sensoren met een zwak uitgangssignaal.
<b>R06</b>	Uit de ondersteunde ingangssignalen van sensoren kan worden gekozen zonder dat hiervoor gewisseld hoeft te worden van firmware.

» Tabel 6.1, requirements bij deze iteratie

Omdat er het risico is bij requirement R06 dat dit de systeemeisen overschrijdt is het ontwerp van deze iteratie in twee delen verdeeld. Het eerste deel (6.1.1) van deze fase richt is gericht op een applicatie die de analoge sensoren via de nieuwe externe ADC ondersteund. Het tweede deel (6.1.2) zal een ontwerp zijn dat bovenop het eerste deel ook ondersteuning biedt voor andere sensoren binnen dezelfde applicatie. Hierin wordt het ontwerp uit het eerste deel gebruikt en wordt deze gedetailleerd aan de hand van een UML klassediagram en sequentiediagram. Bij dit laatste deel is het risico van de systeemeisen aanwezig. Bij de ontwikkelfase zal naar voren komen in hoeverre het ontwerp in 6.1.2 gerealiseerd kan worden.

### 6.1.1 Ondersteuning analoge sensoren

Het belangrijkste doel voor deze applicatie is de ondersteuning van een analoge sensor door middel van een externe ADC (R02). Het gaat in ieder geval om de sensoren met een uitgangssignaal van 4-20 mA, 0-2,5 V, 0-5 V en 0-10 V (R03). Voor het ontwerp heb ik mij eerst de onderstaande ontwerpkeuzes gemaakt.

#### Versterking (R04)

Voor het lezen van het voltage is in de software de versterking de belangrijkste variabele. Om dit extern in te kunnen laten stellen zou deze instelling ontvangen kunnen worden via LoRa. Op de PCB zijn ook DIP-switches beschikbaar. Deze zijn uitleesbaar via de externe ADC. De meest gebruiksvriendelijke oplossing is om de versterking op de DIP-switches in te laten stellen door de eindgebruiker. Daarom heb ik voor dit laatste gekozen. Voor de opdrachtgever was dit goed, al zou het kunnen dat er in de toekomst wel gebruik gemaakt zal moeten worden van LoRa voor extra instellingen.

Wanneer alle versterkingswaardes instelbaar zijn gebruikt dit 8 van de 16 mogelijkheden. De 16 mogelijkheden zijn uitgebeeld in tabel 6.2.

DIP-setting	Functie	DIP-setting	Functie
0000	gain uitgeschakeld	1000	vrij
0001	gain 2	1001	vrij
0010	gain 4	1010	vrij
0011	gain 8	1011	vrij
0100	gain 16	1100	vrij
0101	gain 32	1101	vrij
0110	gain 64	1110	vrij
0111	gain 128	1111	vrij

» Tabel 6.2, Mogelijkheden en functies DIP-switch

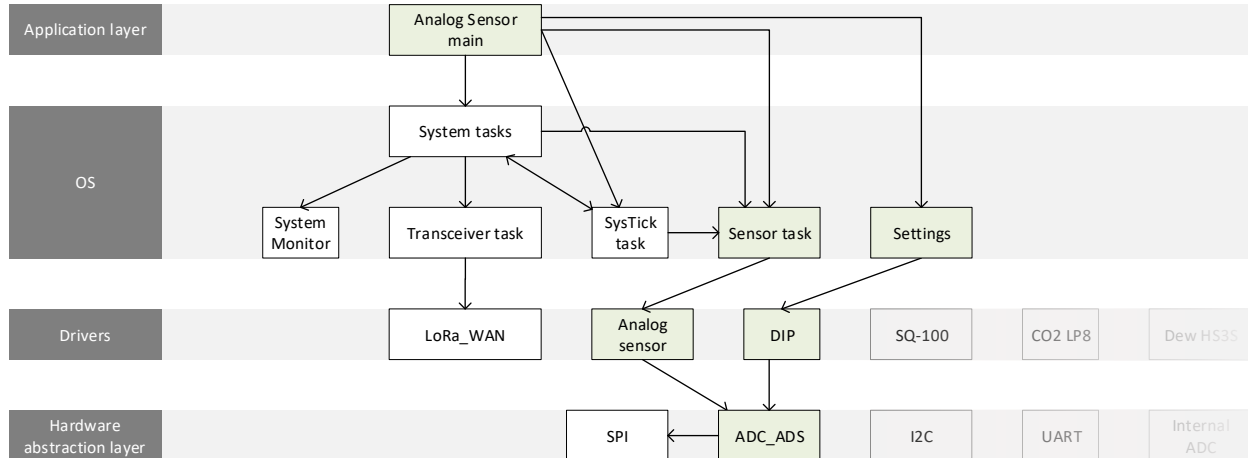
#### Voltages opsturen, berekeningen extern

Voor het versturen van de data van analoge sensoren kan de gemeten spanning in volt opgestuurd worden naar de backoffice. Omdat de software generiek is, is de juiste eenheid die hoort bij de sensor en hoe dit omgerekend moet worden onbekend. De sensornode zou deze gegevens kunnen ontvangen via LoRa bij het opstarten. Een alternatieve oplossing is dat deze berekening opgeslagen is in de backend en de berekeningen hier ook gedaan wordt. Het voordeel hiervan is dat de GEN-node minder berekeningen hoeft te doen, wat voordelig is voor dit soort low power apparaten. Vandaar dat ik voor dit laatste heb gekozen, dit is in overleg goedgekeurd door mijn opdrachtgever.

Tot slot heb ik ervoor gekozen abstracties aan te brengen in het ontwerp. Hierdoor ontstaan nieuwe modules: settings, analog sensor en DIP-switch. Dit zorgt ervoor dat de software niet te afhankelijk wordt van het type hardware dat gebruikt wordt. Ook in het geval van de settings-module kan ervoor gekozen worden om in de toekomst de settings binnen te laten komen door middel van bijvoorbeeld LoRa als de DIP-switch niet genoeg mogelijkheden meer geeft.

De instelbare versterkingsfactoren geven ondersteuning voor sensoren met een maximale uitgangsspanningen van minder dan 2,5V. Sensoren van 0 – 5 V en 0 – 10 V worden geschaald naar de reeks 0 – 2,5 V met behulp van een DC/DC-converter, zoals bepaald in de analysefase. Deze sensoren kunnen gebruik maken van de DIP-optie ‘gain uitgeschakeld’ net als de 0 – 2,5 V sensoren. 4 – 20 mA sensoren kunnen hier ook gebruik van maken, maar moeten wel gebruik maken van de weerstand van 100 ohm. Deze is ingebouwd bij de pin CUR. Overigens is het beter om gebruik te maken van een weerstand van 120 ohm ( $20 \text{ mA} \times 120 \Omega = 2,4 \text{ V}$ ). Dit verhoogt namelijk de precisie van de meting.

Deze ontwerpkeuzes hebben geleid tot het ontwerp in afbeelding 6.1. Dit ontwerp bouwt voort op de bestaande softwarearchitectuur (afbeelding 5.7). De blokken in het groen zijn nieuw of gewijzigd ten opzichte van de al bestaande software.



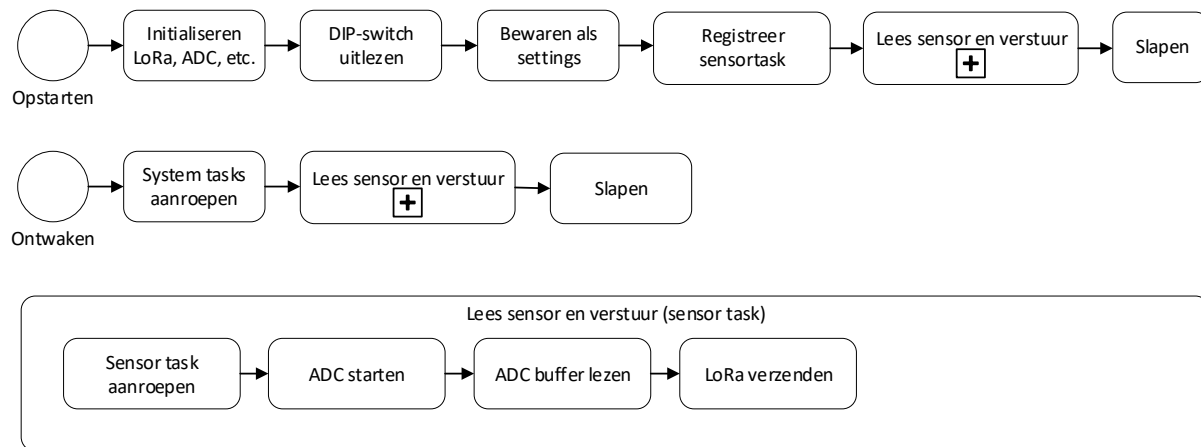
» Afbeelding 6.1, Lagen en onderdelen van de applicatie analog sensor

Dit ontwerp is voor de applicatie “analog sensor”. De algehele software bevat meerdere applicaties, voor elk type sensor. Een applicatie kan gebruik maken van modules uit onderliggende lagen. Uit elke applicatie komt 1 binair bestand die geüpload kan worden naar de hardware.

Een gedetailleerd ontwerp met hierin de nieuwe modules zal volgen in de volgende paragraaf.



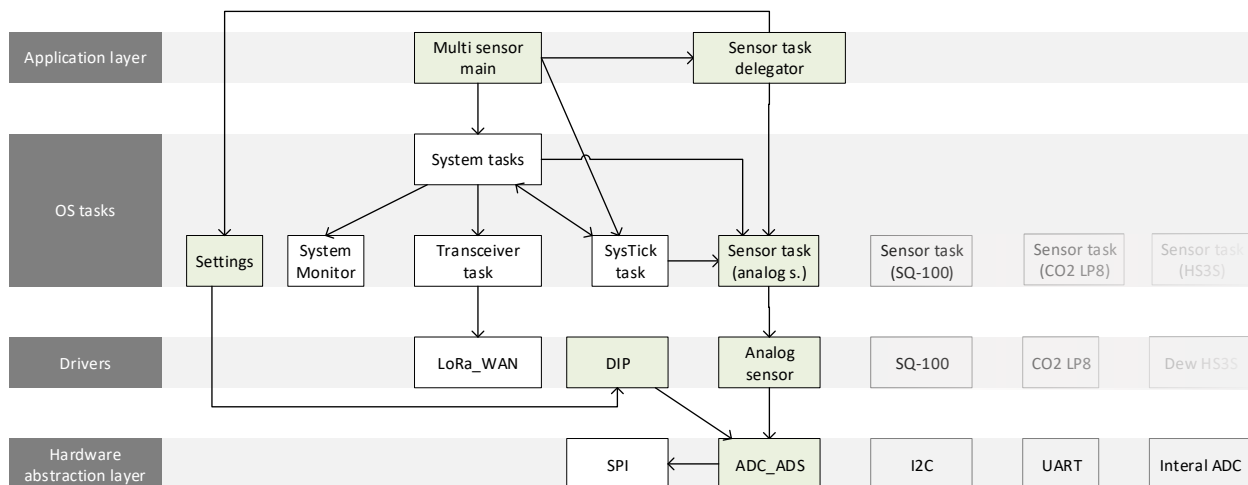
Het proces heb ik inzichtelijk gemaakt in het sequentiediagram in afbeelding 6.2. Hier is te zien dat de DIP-switch eenmalig uitgelezen wordt, alleen bij het starten van het apparaat. Mocht de instelling opnieuw gemaakt moeten worden kan de node gereset worden met behulp van een magneet of door het onderbreken van de voeding. Op deze manier hoeft de de DIP-switch niet regelmatig uitgelezen te worden door de ADC, wat gunstig is in het energieverbruik.



» Afbeelding 6.2, Sequentiediagram analog sensor

### 6.1.2 Ondersteuning gewenste sensoren

De ondersteuning van analoge sensoren is één van de dingen die in een generiekere applicatie moet komen. Deze paragraaf laat het ontwerp zien dat als doel heeft om meerdere soorten ingangssignalen te kunnen kiezen (waaronder analoge sensoren), zonder dat er gewisseld moet worden van firmware (R06). Hiervoor is het ontwerp in de vorige paragraaf uitgebreid. De nieuwe applicatie krijgt de naam multi\_sensor.



» Afbeelding 6.3, Lagen en elementen van de applicatie multi sensor

Ik heb ervoor gekozen om een sensor task delegator te ontwikkelen voor deze applicatie. Deze zal de juiste sensor task registreren bij het systeem aan de hand van de settings vanuit de DIP-switch. Dit moet het namelijk mogelijk maken om ook bestaande sensor\_tasks te integreren in één applicatie.

De 16 mogelijkheden van de DIP-switches krijgen de volgende functies vermeld in tabel 6.3. Links de generieke analoge sensoren en rechts sensoren met een specifiekere interface.

DIP-setting	Sensor	DIP-setting	Sensor
0000	Analoog - Gain uitgeschakeld	1000	3-draads gain 1 (Alphasense NO2/SO2)
0001	Analoog - gain 2	1001	3-draads gain 8 (RTD PT500)
0010	Analoog - gain 4	1010	Ds18b20
0011	Analoog - gain 8	1011	Sh31 Celcius
0100	Analoog - gain 16	1100	LP8 CO2
0101	Analoog - gain 32	1101	Hs3s
0110	Analoog - gain 64	1110	SPS30
0111	Analoog - gain 128	1111	vrij

» Tabel 6.3, Mogelijkheden en functies multi sensor

De tabel is tot stand gekomen in overleg met de begeleider. In dit overleg hebben we gekozen welke sensoren uiteindelijk gebruikt moeten gaan worden in deze applicatie.

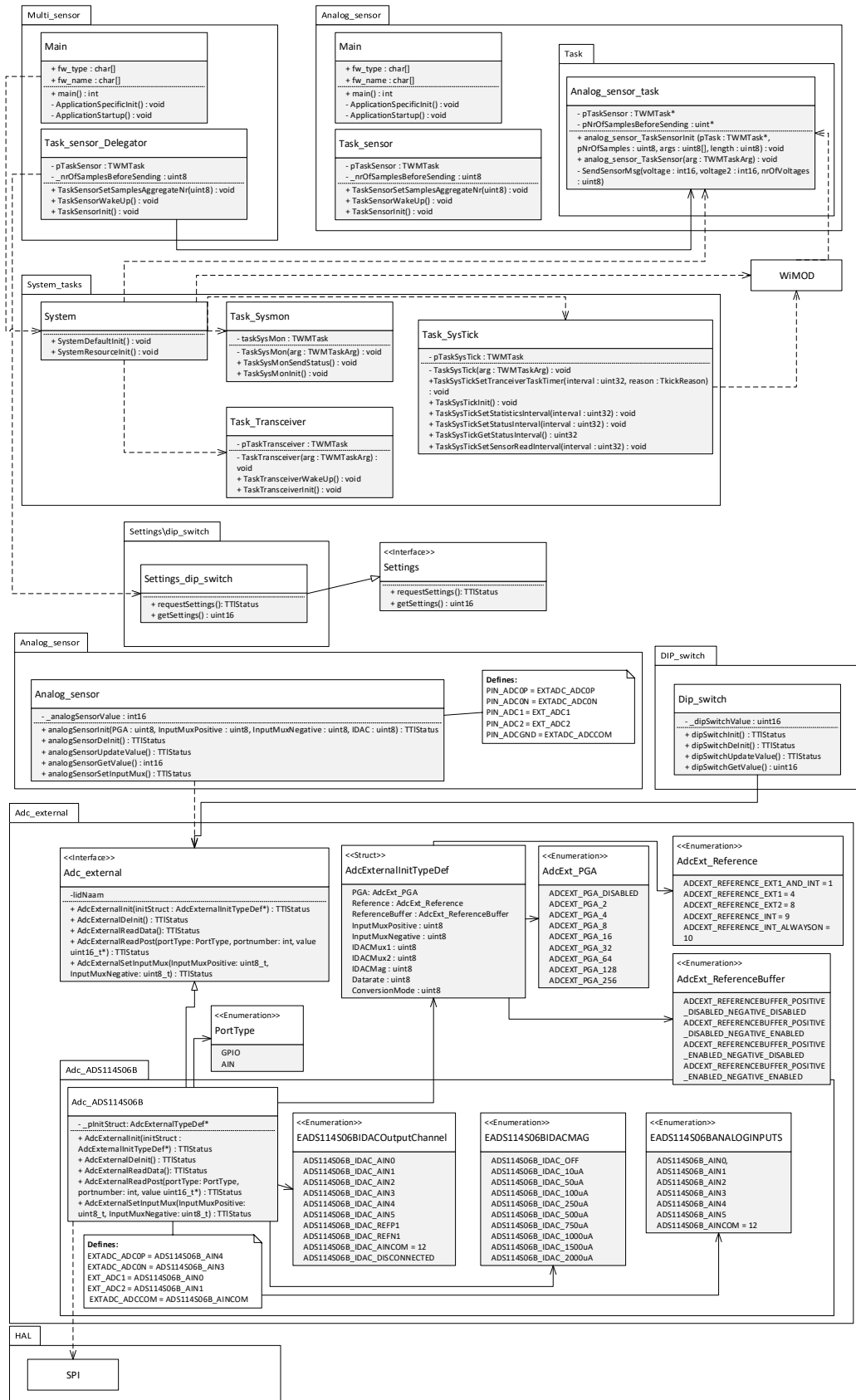
Het processtroomdiagram in afbeelding 6.2 is hier nog steeds van toepassing. Het blokje waarin de sensortask wordt geregistreerd is hiermee een iets omvangrijkere taak geworden aangezien de software nu een keuze moet maken welke sensortask geregistreerd wordt.

## Detailontwerp

Nu de grote lijnen geschetst zijn van het ontwerp heb ik dit verwerkt in een UML klassediagram gemaakt voor de nieuwe applicatie multi\_sensor, in afbeelding 6.4. De programmeertaal C bevat geen klassen, daarom bestaat een klasse in dit ontwerp in werkelijkheid uit een .c/.h-bestand. Een pakket staat hier in werkelijkheid voor een module.

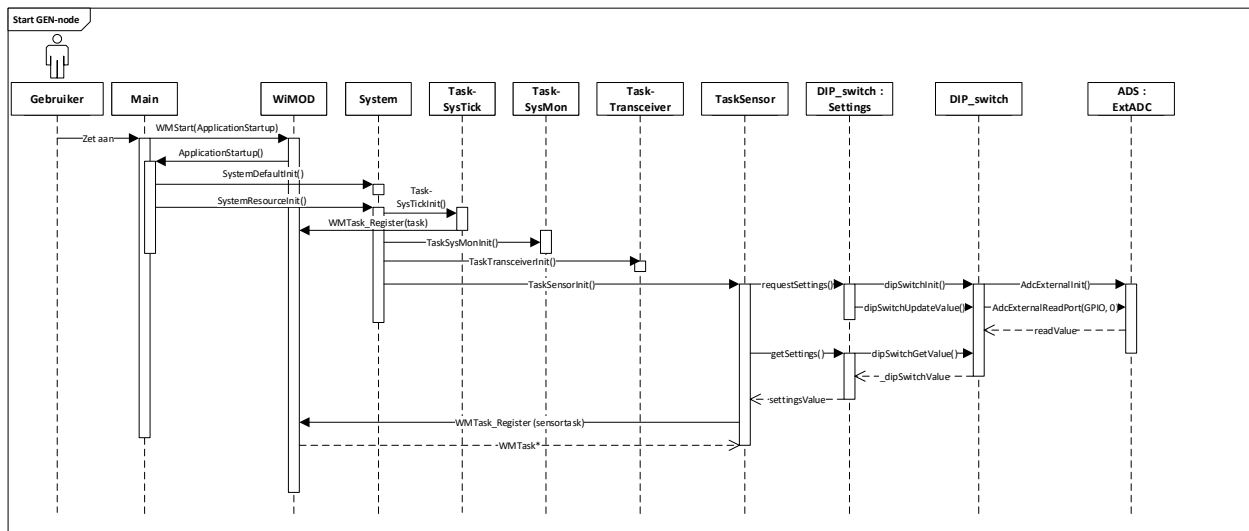
Het toevoegen van een nieuwe sensor (requirement R11) is hiermee een kwestie geworden van het toevoegen van een verwijzing naar de nieuwe sensortaak in Task\_sensor\_delegator.

Met het structtype AdcExternallnitTypeDef kan de manier van uitlezen van een analoge sensor makkelijk worden aangepast (requirement R12), en is de manier van uitlezen onafhankelijk gemaakt van de gebruikte hardware (ADC).

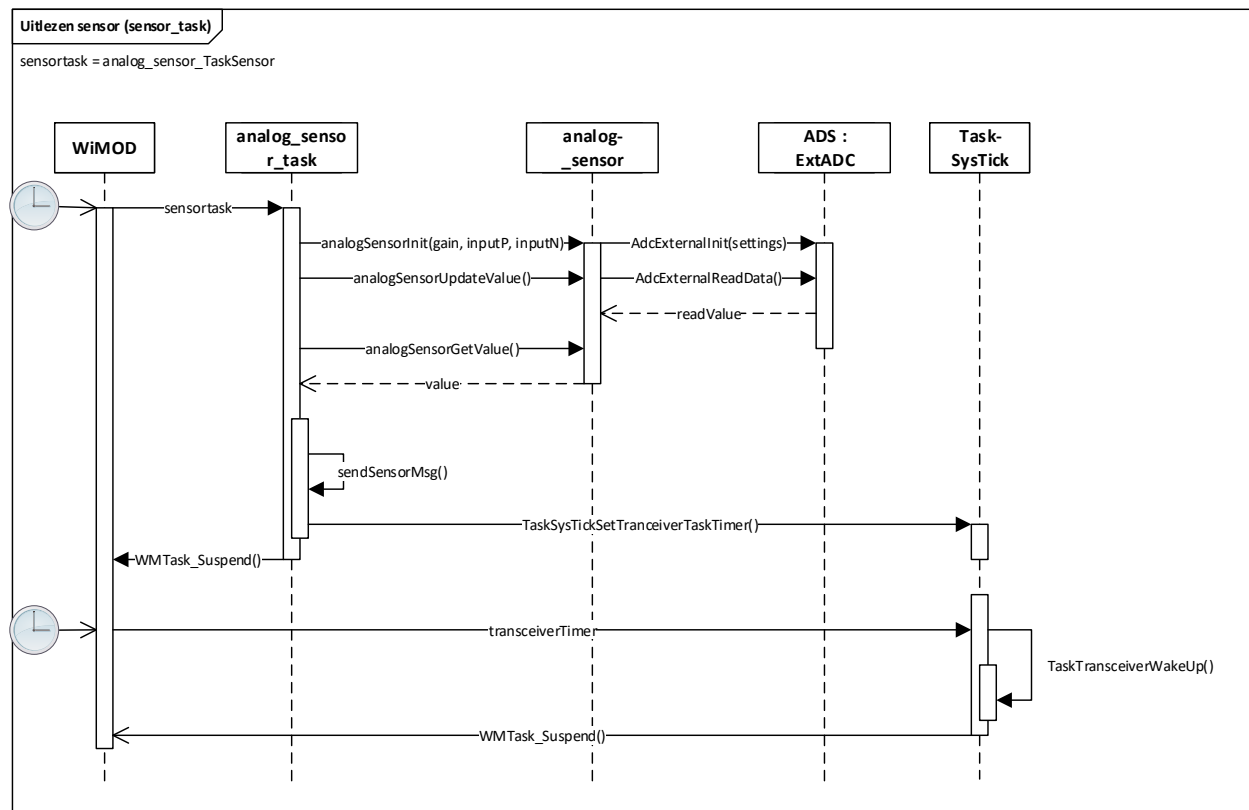


» Afbeelding 6.4, klassediagram van multi\_sensor, gebruikmakend van analog\_sensor

Nu de “klassen” bekend zijn, heb ik UML sequentiediagrammen kunnen maken om het processtroombiagram in afbeelding 6.2 beter te kunnen detailleren.



» Afbeelding 6.5, sequentiediagram starten GEN-node



» Afbeelding 6.6, sequentiediagram uitlezen sensor

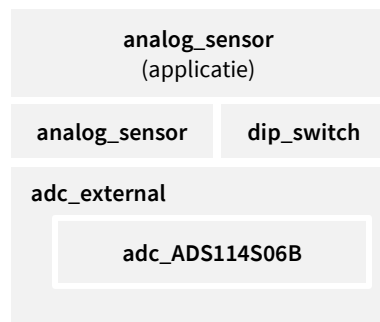
## 6.2 Ontwikkelfase

Oriëntatie-fase	Analyse-fase	Iteratie 1	Iteratie 2	Iteratie 3	Advies-fase
		Ontwerpfase	<u>Ontwikkelfase</u>	Testfase	

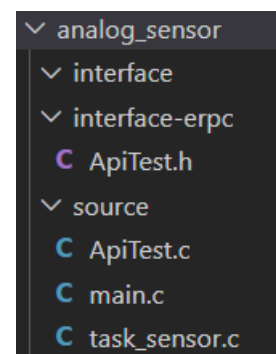
In deze fase wordt de firmware aangepast op basis van het ontwerp in de ontwerpfase. De firmware zal in deze fase ondersteuning krijgen voor analoge sensoren en de externe ADC. Het is een requirement (R06) dat dit in één applicatie gebeurt die de keuze geeft voor andere soorten sensoren.

### 6.2.1 Applicatie analog sensor

De modules in afbeelding 6.7 worden hier behandeld.



» Afbeelding 6.7, Applicatiemodule



» Afbeelding 6.8, Applicatiemodule

### Applicatie

Allereerst heb ik een nieuwe applicatie aangemaakt genaamd `analog_sensor`, zie afbeelding 6.8. Een applicatie bevat de main van de firmware en de onderliggende functies en taken. Belangrijk en het onderscheidende van een applicatie is de `TaskSensor`. Deze schakelt de voeding in van de ADC, roept functies op van analog sensor module en maakt van de teruggekregen waarden een LoRa-bericht.

### Sensormodule - `analog_sensor`

Deze module is nieuw en bevat functies als `analogSensorInit`, `analogSensorUpdateValue` en `analogSensorGetValue`.

Bij initialisatie van de `analog_sensor` wordt er een struct opgebouwd genaamd `AdcExternalInitStructure`. Ik heb gekozen voor een struct omdat dit het overzichtelijk maakt om een externe ADC te configureren en omdat de gehele configuratie makkelijker te delen is door een pointer naar deze struct. Voor het opstellen van de struct heb ik onderzocht welke ADC-instellingen nodig zijn voor de GEN-node. De struct ziet er uiteindelijk zo uit, waarbij de gain en de InputMux (de gebruikte poorten) een variabele is, bepaald op applicatieniveau:

```

AdcExternalInitTypeDef AdcExternalInitStructure;
AdcExternalInitStructure.PGA = gain;
AdcExternalInitStructure.Reference=ADCEXT_REFERENCE_INT;
AdcExternalInitStructure.ReferenceBuffer=ADCEXT_REFERENCEBUFFER_POSITIVE_DISABLE
D_NEGATIVE_DISABLED;
AdcExternalInitStructure.ConversionMode=ADCEXT_CONVERSIONMODE_SINGLESHOT;

AdcExternalInitStructure.InputMuxPositive=InputMuxPositive;
AdcExternalInitStructure.InputMuxNegative=InputMuxNegative;

AdcExternalInitStructure.IDACMux1=ADS114S06B_IDAC_DISCONNECTED;
AdcExternalInitStructure.IDACMux2=ADS114S06B_IDAC_DISCONNECTED;
AdcExternalInitStructure.IDACMag=ADS114S06B_IDAC_OFF;
AdcExternalInitStructure.Datarate=ADS114S06B_DATARATE_50_SPS;
TTLStatus status = AdcExternalInit(&AdcExternalInitStructure);

```

Broncode analog\_sensor.c

Ook geeft dit flexibiliteit wanneer er van type ADC gewisseld wordt. Dezelfde struct kan gebruikt blijven worden, hooguit dat enkele specifieke onderdelen van de configuratie aangepast moeten worden naar dat type ADC. Analog\_sensor maakt gebruik van de module adc\_external.

### Adc\_external

De module adc\_external (afbeelding 6.9) heb ik geïntroduceerd om als het ware een interface te vormen voor een externe ADC. Het bevat een .h-bestand en bevat een onderliggende modules van externe ADC's. In dit geval alleen de ADS114S06B van Texas Instruments.

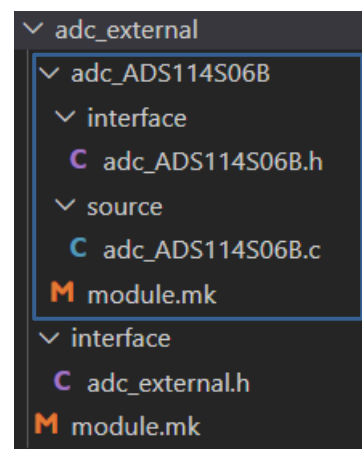
Het .h-bestand adc\_external.h definieert hoe de AdcExternalInitStructure eruitziet en welke functies er aangeroepen kunnen worden op een externe ADC, zoals AdcExternalInit of AdcExternalReadData.

De module.mk verwijst naar de module van de externe adc die in gebruik is, momenteel dus adc\_ADS114S06B. Module.mk is een onderdeel uit de softwarebouwomgeving van Technolution.

### Adc\_external - adc\_ADS114S06B

In adc\_ADS114S06B.c en adc\_ADS114S06B.h wordt er invulling gegeven aan de functies zoals AdcExternalInit, zodat ze specifiek zijn gemaakt voor de huidige ADC.

Op deze manier wordt er invulling gegeven aan de functie AdcExternalReadData door deze ADC:



» Afbeelding 6.9, Modules adc\_external en adc\_ADS114S06B

```

Tl1Status AdcExternalReadData(uint16_t *value) {
    //Send start command for analog -> digital conversion
    while(Adc_ADS114S0B_Start() != TL_STATUS_OK);
    //Sleep until conversion is done. This depends on datarate (SPS).
    TlOsalSleep(24);
    //Send command to retrieve the data
    while(Adc_ADS114S0B_ReadData(value) != TL_STATUS_OK);

    return TL_STATUS_OK;
}

```

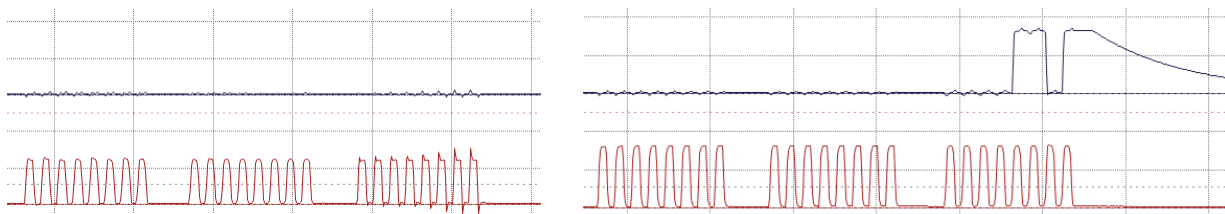
Broncode adc\_ADS114S06B.c

Voorafgaand is AdcExternalInit al uitgevoerd zodat we in deze functie direct van start kunnen met het sturen van het Start-commando voor ADC-conversie (Adc\_ADS114S0B\_Start). Hier is de datarate van de conversie op 50 SPS (samples per second) gezet bij initialisatie, zodat de conversie 20,211 ms duurt [4]. In hoofdstuk 8 zal de uiteindelijke datarate definitief bepaald worden. Deze Met TlOsalSleep(24) wordt op de conversie gewacht waarna er het commando Read Data gegeven kan worden. Deze functie stuurt onderliggend met SPI een byte dat staat voor het commando Read Data en verwacht 2 bytes terug. De SPI-driver bestond al in de software en was al getest voor het gebruik van deze ADC. De functies die het bijbehorende commando sturen waren al wel geschreven, zoals Adc\_ADS114S0B\_ReadData.

### Sensors - dip\_switch

De nieuwe module dip\_switch is op dezelfde manier opgebouwd als de module analog\_sensor. Ook hier is het weer zo dat de manier van opvragen van de waarden van de dip-switch niet verandert als de onderliggende hardware verandert. De module.mk van dip\_switch verwijst naar (en hergebruikt) de AdcExternal-module (requirement R10). Op deze manier wordt er rekening gehouden bij het ontwerp met de beperkte geheugenruimte, wat een kansverlagende maatregel is bij het risico: er is onvoldoende geheugen op de hardware. De nieuw gemaakte module settings fungeert als een interface voor de dip\_switch-sensor.

### Problemen SPI-communicatie



» Afbeelding 6.10, Blauw: ontvangen SPI-signaal van ADC, Rood: SPI-klok. Links is fout.

Er traden problemen op met de communicatie tussen de externe ADC en de microcontroller. Het zorgde voor onvoorspelbaar gedrag waarbij soms de ADC succesvol kon worden uitgelezen. Ten eerste heb ik de SPI-driver geprobeerd te debuggen. Ook heb ik de SPI-

signalen met een oscilloscoop uitgemeten met hulp van mijn begeleider. Dit liet zien dat de ADC vaak geen bytes terugstuurt, zie de linkerkant van afbeelding 6.10. De rechtergrafiek toont het moment van een goede communicatie. Ik ben vervolgens van PCB gewisseld om te kijken of het aan de hardware lag. Dit gaf hetzelfde resultaat. Ik heb toen de voeding van de ADC gecontroleerd. Hieruit bleek het dat deze onvoldoende gevoed werd. Dit bleek achteraf gezien heel eenvoudig opgelost te kunnen worden door in de software het poortje `adc_enable` hoog te zetten. Helaas is één van de twee beschikbare GEN-nodes kapotgegaan tijdens dit onderzoek (GEN 0001). Nu had ik alleen nog GEN 0002, waardoor de kans op het risico “De GEN-nodes raken defect waardoor er niet getest kan worden” groter is geworden aangezien ik nu geen back-up hardware meer heb.

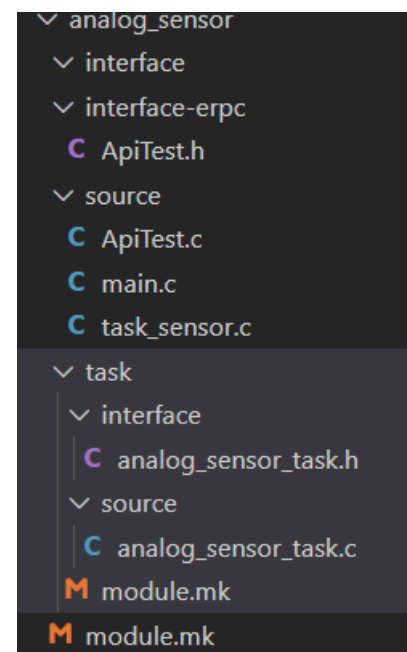
#### 4 – 20 mA en een onjuiste weerstand

De 4-20mA-sensoren leken goed binnen te komen in de GEN-node, behalve dan dat er een kleine afwijking gemeten werd. Zo gaf de SMP6-a pyranometer bijvoorbeeld 244 mV als meetwaarde. Dit laat zich omrekenen naar 2,44 mA (Wet van Ohm:  $\frac{0,244 V}{100 \Omega}$ ). Dit is minder dan de minimale waarde van 4 mA. Ik voerde een weerstandsmeting uit en daaruit bleek dat de 100  $\Omega$  weerstand [1] die is aangebracht in de PCB in werkelijkheid 60  $\Omega$  is. Dat verandert de berekening:  $\frac{0,244 V}{60 \Omega} = 0,00407 A$ . Dat komt dus overeen met 4mA en is de juiste waarde. Hier moet dus rekening mee worden gehouden bij de omrekening van de waardes in de back-office.

#### 6.2.2 Applicatie multi sensor

Voor deze applicatie is het nodig om de specifieke gedeeltes van meerdere (al bestaande) applicaties te kunnen gebruiken in 1 nieuwe applicatie. Omdat elke applicatie ook grotendeels dezelfde, niet-specifieke onderdelen heeft, zoals de main en het starten van het OS, moeten de applicaties in tweeën worden gesplitst.

In afbeelding 6.11 is te zien hoe de applicatiemodule `analog_sensor` eruitziet nadat ik deze gesplitst heb. Het lichtgrijze gedeelte geeft de nieuwe module (task) aan binnen de applicatiemodule. Hierin staat hoe de sensor op applicatieniveau uitgelezen moet worden. Vanuit de nieuwe applicatie `multi_sensor` kunnen we nu verwijzen naar deze `task-module`, zonder de complete applicatie-module te gebruiken.



» Afbeelding 6.11, Tweedeling `analog_sensor` applicatiemodule



Aan de hand van een switch-statement in de applicatie multi\_sensor kan ik nu de benodigde task uit de task-module registreren op basis van de (DIP-switch-)settings uit de settings-module.

### RAM-problemen

Tijdens het schrijven van de software heb ik problemen gehad door het overschrijden van de systeemeis: maximaal 32 kB RAM. Allereerst was er een sprake van een stack overflow. Onvoldoende ruimte op de stack dus. Dit bleek te komen door een fout in het linker-script, hier wees mijn opdrachtgever mij op. Dit .ld-bestand vertelt hoe het geheugen ingedeeld wordt op de microcontroller. Hieronder is een fragment van dit script te zien.

```
/* Specify the memory areas */
MEMORY {
    FLASH (rx)      : ORIGIN = 0x08000000, LENGTH = 0x0001FFFF
    RAM (xrw)        : ORIGIN = 0x20000000, LENGTH = 0x00003FE0
    assert_pc (rwx) : ORIGIN = 0x20003FE4, LENGTH = 0x00000014 /* storage
for assert PC */
    boot_cnt (rxw)   : ORIGIN = 0x20003FF8, LENGTH = 0x00000004 /* storage
for boot counter */
    sr_reason (rwx) : ORIGIN = 0x20003FFC, LENGTH = 0x00000004 /* storage
for soft reboot reason */
    MEMORY_B1 (rx)   : ORIGIN = 0x60000000, LENGTH = 0K
    /* EEPROM (rx)      : ORIGIN = 0x08080000, LENGTH = 0x00000FFF DO
NEVER ERASE THE EEPROM this is where the device EUI is stored */
}
```

Fragment uit stm32l1xx\_flash.ld

Het vetgedrukte deel laat de grootte van het RAM-gedeelte zien. In dit gedeelte staat ook de stack. 0x3FE0 staat voor 16.352 bytes (16 kilobytes). Echter is er op deze microcontroller 32 kilobytes RAM beschikbaar [11]. Ik heb daarom de lengte aangepast naar 0x00007FE0 en de erop volgende adressen verplaatst. Hierdoor is het RAM-gedeelte verdubbeld waarmee nu ook het stackprobleem is opgelost.

### Flash-problemen

Het volgende probleem dat opduikt naarmate er meer sensorapplicaties worden toegevoegd aan de multi\_sensor-applicatie is het flashgeheugen. Er is 128 kilobyte vrij voor de firmware (systeemeis). Om aan alle opgegeven sensoren te voldoen is er zo'n 136 kilobyte nodig, blijkt nu. Dit probleem is één van de beschreven risico's (Er is onvoldoende geheugen op de hardware). De kansverlagende maatregel heeft onvoldoende effect gehad (rekening houden met het tekort bij het ontwerp). Daarom pas ik de impactverlagende maatregel toe: het schrappen van mindergebruikte software delen. Om dit te bereiken heb ik mij eerst gefocust

op de statische bibliotheek. Een groot gedeelte van de firmware bestaat uit deze bibliotheek van de fabrikant van de LoRa-module. De bibliotheek is niet vrij beschikbaar. Ik heb de fabrikant gevraagd of er een lichtgewicht versie is van deze bibliotheek. De fabrikant meldt dat de bibliotheken juist groter zijn geworden vanwege de ontwikkelingen rondom LoRaWAN en nieuwe functionaliteiten. Ook wordt er gemeld dat er een nieuwe module wordt ontwikkeld genaamd iM881A-XL. Deze module bevat 192 kilobyte flashgeheugen en wordt verwacht in het tweede of derde kwartaal van 2020. Dit is een plan B. Geen ideale oplossing voor nu aangezien dit te veel tijd zou kosten.

Daarom besloot verder te gaan met functionaliteiten schrappen, ditmaal in de Technolutionsoftware. Het schrappen van eRPC (embedded Remote Procedure Call) zorgt voor een vermindering van 6 kilobyte. Het schrappen van informatieve logging op de UART, waardoor nu alleen nog fouten getoond worden op de console zorgt eveneens voor een vermindering van zo'n 6 kilobyte. Het eerste wordt niet heel actief gebruikt momenteel en het tweede is alleen nodig voor debuggen. De opdrachtgever is hiermee akkoord gegaan. In de toekomst kan er gekozen worden voor de grotere module waar meer flashgeheugen bij aanwezig is.

### Overige problemen

Tot slot werkte de 3,3 volt-uitgang niet naar behoren voor de bestaande sensoren. Het bleek dat de aansturing van deze GPIO-poort veranderd was van normally closed naar normally open. Dit heb ik opgelost door dit in de software steeds aan te passen op precompiler-niveau voor het geval dat de nieuwe PCB wordt gebruikt, maar zodanig dat de bestaande software nog steeds op oudere PCB-ontwerpen kan worden gebruikt. Ook moest een deel van de bestaande SH31-sensorfirmware aangepast worden omdat deze nog alleen ontwikkeld was voor een oudere PCB met een andere hardware-interface.

## 6.3 Testfase

Oriëntatie-fase	Analyse-fase	Iteratie 1	Iteratie 2	Iteratie 3	Advies-fase
		Ontwerpfase	Ontwikkelfase	Testfase	

Deze fase bestaat uit het testen van de requirements die behoren bij deze iteratie. Ik heb in de bijlage Testrapport een testaanpak opgesteld. In deze aanpak staat dat ik voor metingen gebruikmaak van een oscilloscoop en hoe ik die gebruik. Ook heb ik in het Testrapport een testplan opgesteld. Hierin wordt de testprocedure beschreven en de verwachte uitkomst. Vervolgens heb ik de testprocedure uitgevoerd. De opdrachtgever heeft geverifieerd dat bij een succesvolle test ook daadwerkelijk voldaan is aan de bijbehorende requirement. Dit proces is op dezelfde manier uitgevoerd in de overige testfases in hoofdstuk 7.3 en 8.3.

**Beschrijving:** De firmware heeft een implementatie voor het aansturen en het uitlezen van de externe ADC.

**R02**

**Precondities:** UART wordt uitgelezen, DIP-switch staat op 0000

**Acties:**

- Sluit 1400 mV aan op de poort ADC0P en de negatieve pool op ADC0N
- Zet de GEN-node aan

**Verwacht resultaat:**

Console geeft: **New sample from sensor: 1400mV**

**Daadwerkelijk resultaat:**

Console geeft 1399 mV

**OK** **Ja**

De hierop volgende tests worden kort behandeld. In de bijlage Testrapport is het volledige rapport te zien inclusief procedure.

**Beschrijving:** De firmware geeft de keuze voor het gebruik van analoge sensoren, zodat er ondersteuning is voor een sensoren met een uitgangssignaal van 4-20 mA, 0-2,5 V, 0-5 V en 0-10 V.

**R03**

**OK** **Ja**

**Beschrijving:** De firmware geeft de keuze voor het toepassen van de benodigde gain, zodat er ondersteuning is voor analoge sensoren met een zwak uitgangssignaal.

**R04**

**OK** **Ja**

Opmerking: De firmware handelt dit goed af, maar de ADC op GEN-node met typenummer 0002 voert de versterking te licht uit.

**Beschrijving:** Uit de ondersteunde ingangssignalen van sensoren kan worden gekozen zonder dat hiervoor gewisseld hoeft te worden van firmware.

**R06**

**OK** **Ja**

Opmerking: Hierbij vervalt wel de eRPC-functionaliteit en de informatieve logging op de UART wegens gebrek aan flashgeheugen.

### **Conclusie**

Met het uitvoeren van deze tests is aangetoond dat er aan de requirements die behoren bij deze iteratie voldaan is. Er zijn wel twee kanttekeningen. Ten eerste wordt R04 juist uitgevoerd in de firmware, maar niet goed genoeg op de GEN-node 0002. Ten tweede vervalt de eRPC-functionaliteit en informatieve logging bij R06. De opdrachtgever heeft aangegeven dat dit niet erg is.

## 7. Iteratie 2: Luchtkwaliteit

De deelvraag van deze iteratie is: Hoe kunnen sensoren ten behoeve van het meten van luchtkwaliteit worden ondersteund? Het gaat hierbij om sensoren die in overleg met opdrachtgever en begeleider gekozen worden. Hieruit is de keuze gevallen op een NO<sub>2</sub>-sensor en een SO<sub>2</sub>-sensor. Deze waren al eens gekocht door Technolution maar nog niet toegepast op de GEN-node.

### 7.1 Ontwerpfase

Oriëntatie-fase	Analyse-fase	Iteratie 1	Iteratie 2	Iteratie 3	Advies-fase
		Ontwerpfase	Ontwikkelfase	Testfase	

In de analysefase is naar voren gekomen wat de types zijn: NO<sub>2</sub>-B43F en SO<sub>2</sub>-B4 van Alphasense. Dit zijn driedraads analoge sensoren. De ontwerpfase moet duidelijk maken hoe deze sensoren toegepast kan gaan worden op de GEN-node. Tevens wordt er tijdens deze iteratie gewerkt aan de ondersteuning van een andere driedraads analoge sensor: de RTD, zoals de PT500. Hiermee is de temperatuur te meten, wat praktisch kan zijn bij kalibreren.

De requirements die de nadruk krijgen tijdens deze iteratie zijn te vinden in tabel 7.1.

R05	De firmware heeft ondersteuning voor geselecteerde driedraadssensoren (NO <sub>2</sub> en SO <sub>2</sub> ).
R07	De firmware heeft de keuze een kleine uitgangsstroom in te schakelen van 0,5mA waarmee de PT500-sensor gevoed kan worden.

» Tabel 7.1, requirements bij deze iteratie

#### 7.1.1 Alphasense gassensoren

##### Sensor: schakeling en behuizing

De NO<sub>2</sub>-B43F en de SO<sub>2</sub>-B4 zijn aangesloten op een schakeling: de Alphasense ISB. Dit bord wordt gevoed door een converter die 3,3 V opschaaft naar 5 V. Dit is gemonteerd in een waterbestendige behuizing met 4 aansluitdraden. Zo kan met 3V3\_OUT en GND op de GEN-node de V<sub>IN</sub> pinnen op het sensorbord gevoed worden. Deze pinnen zijn te zien in afbeelding 7.1.



» Afbeelding 7.1, Aansluitingen Alphasense Individual Sensor Board (ISB) [7]

Typend bij deze sensoren is dat er twee keer gemeten moet worden. Hiervoor gaan de ingangen ADC2 en ADC1 voor gebruikt worden. ADC2 zal in verbinding staan met de positieve pool van OP1, dit is de Working Electrode (WE). ADC1 zal in verbinding staan met de positieve pool van OP2, dit is de Auxiliary Electrode (AE). De WE is het signaal dat reageert op NO<sub>2</sub> of SO<sub>2</sub>. En AE is het signaal dat de hier niet op reageert maar wel kenbaar maakt wat de weerstand is die de sensor en de draden oplevert. Om erachter te komen wat de hoeveelheid gemeten gas is in parts per billion (ppb), moet er gerekend worden met de spanningswaardes van WE en AE, maar ook met enkele constanten die vanuit de fabriek voor elke sensor specifiek zijn bepaald.

### Software (R05)

De Alphasense-sensoren kunnen toegepast worden door een uitbreiding van de analog\_sensor-taak. Door bij de initialisatie van de bijbehorende taak een modus mee te geven kunnen de bijbehorende instellingen voor de ADC worden uitgevoerd. Die instellingen zijn: twee keer meten: AIN1 (ADC2) en AIN0 (ADC1), versterkingsfactor 1x en 3v3\_out aan tijdens het uitlezen.

Het LoRa-bericht dat verstuurd wordt bevat de metingen van de WE en de AE in millivolt. Zo kan de omrekening of kalibratie in de backoffice gebeuren per sensor. Dit bericht is verder gespecificeerd in de bijlage Ontwerprapport in hoofdstuk 3.

### Formule

Vervolgens heb ik onderzocht hoe millivolts omgerekend kunnen worden naar parts-per-billion (ppb). De standaard omrekenformule naar ppb is hieronder te vinden: [12]

$$NO_2 [ppb] = \frac{(V_{WE} - V_{WE,zero}) - n(V_{AE} - V_{AE,zero})}{Sensitivity}$$

Dit is de eenvoudige formule. Hierbij zijn de spanningen in mV.  $V_{WE,zero}$ ,  $V_{AE,zero}$  en  $Sensitivity$  zijn te vinden in de meegeleverde documenten vanuit de fabriek, dit is voor elke sensor anders.  $n$  is de factor afhankelijk van de temperatuur, deze is te vinden in de tabel van bovenstaande bron.

### Kalibratie

Door elektronische afwijkingen kan het nodig zijn de nulafwijking (zero) te kalibreren. Dat gebeurt aan de hand van deze vergelijking: [12]

$$(V_{WE} - V_{PCBWE}) - n(V_{AE} - V_{PCBAE}) = 0$$

Hierbij is  $V_{PCBWE}$  en  $V_{PCBAE}$  de elektronische (nul)afwijking op de ADC.

Naast temperatuur is de NO<sub>2</sub>-B43F ook gevoelig voor O<sub>3</sub>. Volgens het RIVM moet daarom de O<sub>3</sub>-waarde ook meegenomen worden in de kalibratie [13]. Daarnaast is de sensor ook gevoelig voor de luchtvochtigheid. In de ideale situatie wordt daarmee dus ook gekalibreerd [14] [15]. De sensor gaat één tot twee jaar mee, maar zal na enkele maanden al in gevoeligheid afnemen. Maandelijks herkalibreren wordt geadviseerd [16].

Voor de SO<sub>2</sub>-B4 is een kalibratie afhankelijk van de temperatuur en de nulafwijkingen voldoende. [7]

### **Apart meten van WE- en AE-signalen**

In eerste instantie was mijn opzet dat de Alphasense-sensoren redelijk op dezelfde manier uitgelezen konden worden als een RTD (zoals te lezen in hoofdstuk 5.2.2), dat ook een driedraadssensor is. In dat geval zou de sensor gebruik maken van de poorten ADC<sub>ON</sub> en ADC<sub>OP</sub>. Door beide uitgangsdraden (WE en AE) hierop te aan te sluiten zou het verschil tussen beiden poorten het resultaat van de meting worden. Echter, omdat dit verschil analoog wordt bepaald is in de software niet meer te achterhalen wat de waardes zijn van de twee uitgangsspanningen los van elkaar. Mijn begeleider vroeg zich af of dit gewenst is voor deze sensoren. Aan de hand van aanvullend onderzoek naar praktijkvoorbeelden en kalibratieformules kwam ik erachter dat het wél nodig was om zowel het WE-draad als het AE-draad los van elkaar te meten. Dit is namelijk nodig in de standaard omrekenformule (zie de paragraaf formule), vanwege de temperatuursafhankelijke factor  $n$  die geldt op de AE. Helaas ontbrak het document [12] van Alphasense waarin deze formule staat op de website van Alphasense. Vandaar dat ik deze informatie en het bestaan van dat document pas later vond.

### **7.1.2 PT500**

Voor de PT500 sensoren is er een kleine stroom nodig. In het vorige project is geadviseerd dit op 0,5mA te zetten. [1]

De PT500 heeft een weerstand van 460-620  $\Omega$  in normale omstandigheden (-20 tot 60 graden Celcius). Dat betekent met 0,5 mA een uitgang tussen de 0,23 V en 0,31 V (wet van Ohm). Hierbij past een versterkingsfactor van 8x (of 4x bij extreme temperaturen). De Full Scale Resolution (FSR, het maximaal meetbare verschil tussen de 2 polen) is bij deze versterking namelijk 0,313 V. Daarnaast is het een driedraadssensor, alleen hoeft hierbij niet 2x gemeten te worden aangezien beide te meten polen dezelfde versterkingsfactor hebben. Het gebruik van ADC<sub>OP</sub> en ADC<sub>ON</sub> als input is voldoende. Het resultaat is het verschil tussen de draadsweerstand en de sensor inclusief de draadsweerstand.

Door wederom gebruik te maken van de task `analog_sensor` en het toepassen van een nieuwe modus kan ook de RTD zoals de PT500 worden gebruikt. Deze modus zal dus de versterking op 8 zetten, de positieve inputMux op ADC0P en negatieve inputMux op ADC0N. Tot slot zullen de IDAC's (stroombronnen) geactiveerd moeten worden met een stroom van 0,5mA vanaf de externe ADC (R07).

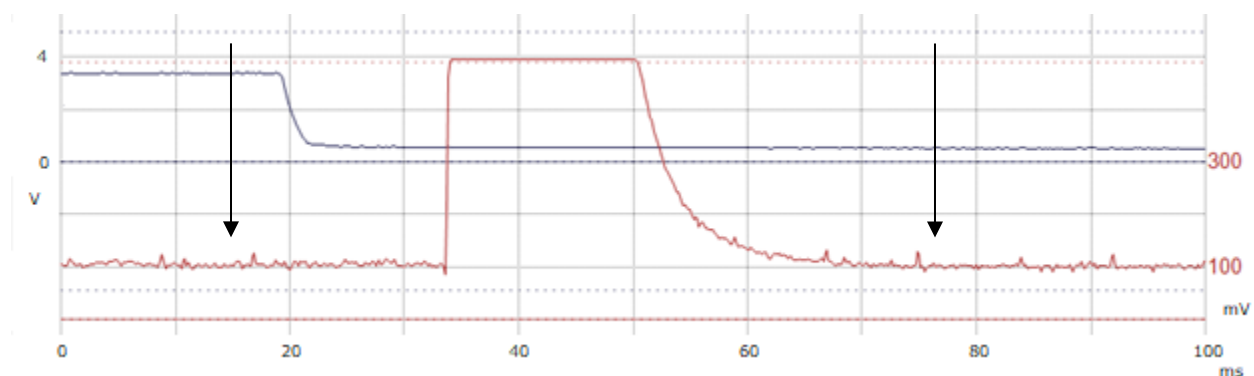
## 7.2 Ontwikkelfase

Oriëntatie-fase	Analyse-fase	Iteratie 1	Iteratie 2	Iteratie 3	Advies-fase
		Ontwerpfase	Ontwikkelfase	Testfase	

Voor de initialisatiefunctie van de sensortaak `analog_sensor` heb ik een argument toegevoegd die de modus selecteert voor een gewone analoge sensor, een Alphasense gassensor of de RTD-sensor. Dit heb ik zo gedaan omdat dezelfde onderliggende modules worden gebruikt en deze bijna op een soortgelijke manier worden aangesproken. Dit bespaart in tegenstelling tot 3 aparte applicaties en onderliggende taak enkele kilobytes flashgeheugen.

### Opstarttijd sensor

Ik ondervond problemen met de meting van de Alphasense sensoren. Het probleem bleek te zijn dat wanneer de sensor enkele milliseconden gevoed wordt (de tijd waarin de periodieke meting gebeurt), de uitgangsspanning daar gelijk is aan die van een ongevoede sensor. In afbeelding 7.2 is te zien hoe de voeding van de sensor uitgaat (blauwe lijn) en WE-uitgang van de Alphasense (rode lijn) na de piek hetzelfde spanningsniveau heeft van 100 mV.



» Afbeelding 7.2, Blauwe lijn: voedingsspanning, rood: uitgang WE Alphasense-sensor NO2-B43F

Door vervolgens te meten met een constante voeding bleek dat na 20 seconden de uitgangsspanning van de sensor redelijk op het gewenste niveau kwam. Maar na zo'n 65 seconden voeden lijkt de uitgangsspanning pas echt gestabiliseerd te zijn. Vanwege dit



verschijnsel heb ik ervoor gekozen om de Alphasense-sensoren permanent te voorzien van voeding vanuit de GEN-node.

### Sensortaak analog\_sensor

De al bestaande sensortaak analog\_sensor (beschreven in hoofdstuk 6.2.1) wordt dus uitgebreid in deze fase, zoals vermeld in het ontwerp. Door bij de initialisatie vanuit de multi\_sensor-applicatie een modus mee te geven als argument kan worden gekozen voor 3 varianten van de sensortaak. De eerste modus is de al bestaande modus, voor gewone analoge sensoren. De tweede modus is voor Alphasense-sensoren en de derde modus voor de RTD (PT500). Hiermee wordt bepaald in de sensortaak welke ingangen gemeten moeten worden, of de IDAC moet worden geactiveerd, of de 3,3 V-uitgang aan blijft na de meting en of er nog een tweede keer gemeten moet worden. Tot slot worden één of meerdere meetwaarden ingepakt in een LoRa-bericht zodra deze zijn gemeten.

## 7.3 Testfase

Oriëntatie-fase	Analyse-fase	Iteratie 1	Iteratie 2	Iteratie 3	Advies-fase
		Ontwerpfase	Ontwikkelfase	Testfase	

Deze fase bestaat uit het testen van de twee requirements die tijdens deze iteratie behandeld worden.

**Beschrijving:** De firmware heeft ondersteuning voor geselecteerde driedraadssensoren (NO<sub>2</sub> en SO<sub>2</sub>).

**R05**

**Precondities:** UART wordt uitgelezen, DIP-switch staat op 1000, alternatieve spanningsmeetmethode aangesloten

**Acties:**

- Sluit een NO<sub>2</sub>-B43F of SO<sub>2</sub>-B4 aan op de poorten beschreven in het Ontwerprapport
- Zet de GEN-node aan
- Wacht op de 2<sup>e</sup> meting.

**Verwacht resultaat:**

Console geeft het meetresultaat in mV en is kloppend

**Daadwerkelijk resultaat:**

De gemeten waarde in de console is vergelijkbaar met de gemeten waarde op de multimeter

**OK Ja**

**Beschrijving:** De firmware heeft de keuze een kleine uitgangsstroom in te schakelen van 0,5mA waarmee de PT500-sensor gevoed kan worden.

**R07**

**Precondities:** UART wordt uitgelezen, DIP-switch staat op 1001

**Acties:**

- Sluit een RTD of soortgelijke weerstand aan op de poorten ADC0N (L1), ADC0P (L2) en REFIN (L3).
- Zet de GEN-node aan

**Verwacht resultaat:**

Console een meetresultaat in mV dat overeenkomt met de uitgangsspanning van de sensor

**Daadwerkelijk resultaat:**

Console geeft een meetresultaat in mV dat vergelijkbaar is met de gemeten waarde op de multimeter.

**OK**   **Ja**

**Conclusie**

De requirements die bij deze iteratie horen hebben ervoor gezorgd dat de spanningen van de Alphasense-sensoren en de PT500-sensoren gemeten kunnen worden op de GEN-node. De opdrachtgever was hiermee tevreden.

## 8. Iteratie 3: Energieverbruik

De deelvraag die bij deze iteratie hoort is: “Hoe kan er rekening worden gehouden met energiezuinigheid?”. Al bestaande en tijdens de vorige iteraties ontwikkelde firmware waar nodig aangepast ten behoeve van de energiezuinigheid. Het is de laatste iteratie voordat de firmware wordt opgeleverd. Daarom wordt er ook de bestaande software netter gemaakt als dit nodig is, zoals drivers. Zodoende is de functionele requirement voor deze iteratie: R01, zie tabel 8.1.

R01	De firmware heeft driverondersteuning voor de ingangen Analog In, SPI en I2C en de uitgangen Analog Out en Digital Out.
R09	Het energieverbruik is lager dan het huidig verbruik wanneer de GEN-node wordt toegepast voor het uitlezen van analoge sensoren met de externe ADC.”

» Tabel 8.1, Requirement bij iteratie 3

Ook belangrijk tijdens deze iteratie is de niet-functionele requirement R09: “Het energieverbruik is lager dan het huidig verbruik wanneer de GEN-node wordt toegepast voor het uitlezen van analoge sensoren met de externe ADC.”.

Hoe laag het energieverbruik precies moet zijn is niet vastgesteld als harde eis. De reden hiervoor is dat dit van tevoren lastig is vast te stellen, zonder dit uitgebreid te hebben onderzocht. De uitkomsten van de onderzoeken en metingen die hiervoor worden gedaan geven kosten en baten. Het behalen van een geëist energieverbruik zou het risico meebrengen dat de baten (een bepaald niveau energieverbruik behalen) niet opwegen tegen de kosten (functionaliteiten moeten schrappen om dat niveau te halen). Daarom worden de kosten en baten afgewogen met de opdrachtgever en begeleider om tot een ontwerp te komen. Dit proces is te lezen in hoofdstuk 8.1 Ontwerpfase.

Het energieverbruik tijdens het uitlezen van een analoge sensor met de externe ADC zal in ieder geval lager moeten zijn dan het geval was voor aanvang van dit project (R09). Wat het verbruik toen was is opgenomen in hoofdstuk 5.2.3.

### 8.1 Ontwerpfase

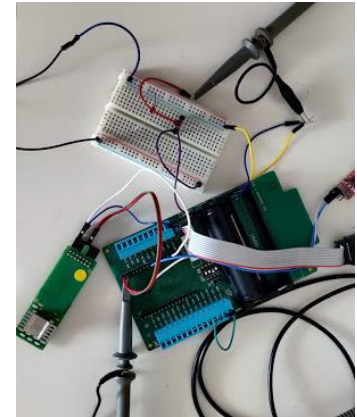
Oriëntatie-fase	Analyse-fase	Iteratie 1	Iteratie 2	Iteratie 3	Advies-fase
		Ontwerpfase	Ontwikkelfase	Testfase	

In de analysefase in hoofdstuk 5.2.3 zijn onderwerpen verzameld die mogelijk kunnen leiden tot een lager energieverbruik. Deze onderwerpen heb ik onderzocht aan de hand van

stroommetingen en aanvullend literatuuronderzoek (zie ook bijlage Ontwerprapport). Hieruit is een ontwerp gekomen.

### Meetmethode

Voor het meten van de stromen heb ik gebruik gemaakt van een oscilloscoop, zoals beschreven in de bijlage Testrapport. De opstelling is te zien in afbeelding 8.1. Ik meet de stroom tussen de negatieve pool van de batterijhouder en de negatieve pool van de batterij. Door een kleine weerstand is de stroom meetbaar geworden als spanning. De weerstand moet zodanig klein zijn dat ik gebruik heb gemaakt van draadweerstand. Door het thuiswerken waren de mogelijkheden voor precisieweerstanden beperkt. Mijn begeleider heeft mij bij het maken van de opstelling geholpen.

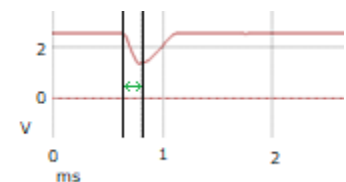


» Afbeelding 8.1, Meetopstelling

### Processortijd: terugkeren naar het OS

Volgens mijn begeleider zou het veel tijd kosten om tijdens het uitvoeren van een taak steeds terug te gaan naar het OS. Een langere processortijd betekent meer energieverbruik. Momenteel wordt een taak herhaaldelijk aangeroepen vanuit het OS totdat deze klaar is. Gebeurt dit niet, dan gebeurt er een watchdog timer overflow, wat zorgt voor een herstart van de node.

Om zeker te weten dat het klopt dat het terugkeren naar het OS lang zou duren heb ik testmetingen uitgevoerd. Uit metingen blijkt dat de tijd dat het kost 0,1 ms is, zie afbeelding 8.2. Door een eenvoudige applicatie te maken waarbij een pin (rode lijn) omlaag wordt gebracht wanneer de taak terugkeert naar het OS en de pin omhoog te brengen bij het terugkeren naar de taak wordt de tijd die hiertussen zit inzichtelijk.



» Afbeelding 8.2, Terugkeertijd

Het alternatief is om grote delen van de software te herschrijven op een manier dat er niet tijdens het uitvoeren van een taak wordt teruggekeerd naar het OS. Dit kan door een met enige regelmaat een watchdog trigger uit te voeren, zodat de watchdog timer niet overloopt. Het uitvoeren van deze trigger blijkt uit metingen eveneens 0,1 ms te kosten. De conclusie is dat hierin geen winst valt te behalen, in tegenstelling tot wat voorheen werd aangenomen.

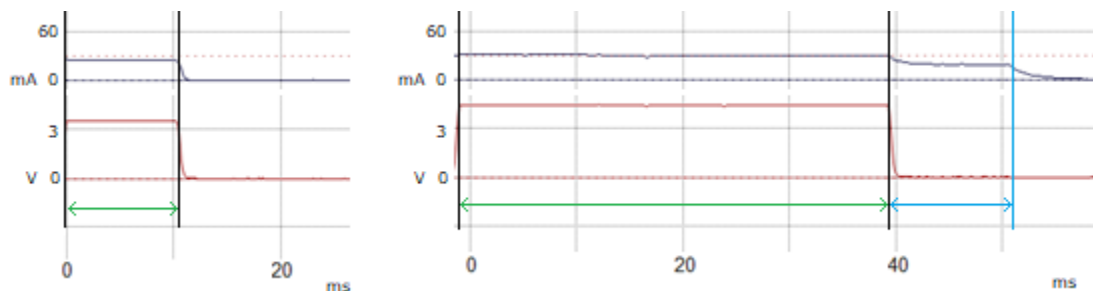
Ook is 0,1 ms relatief kort. Daarom heb ik ervoor gekozen om geen aanpassingen te doen aan de software op dit vlak.

## Processortijd: logging

Eén van de functionaliteiten die processortijd kost, terwijl het in praktijk alleen uitgelezen wordt in testomgevingen, is de UART-log op de console. Ik kwam hierachter door stroom te meten met en zonder log.

Ik heb hiervoor een testapplicatie gemaakt die 10ms lang de testtaak actief houdt. Dit kost 11ms, zie ook afbeelding 8.3. Vervolgens heb de taak aangepast zodat na het uitvoeren van de while-loop die 10ms duurt ook 4 logberichten te printen. Dit kost ruim 40ms (groen) en een uitloop van 12 ms (blauw) na het uitvoeren van de gehele taak. Dat komt neer op een oppervlakte (stroom  $\times$  tijd) van zo'n 0,00039 mAh in tegenstelling tot 0,00008 mAh zonder logging.

Aangezien deze informatieve logging alleen nodig is met debuggen is ervoor gekozen om dit in firmware voor productieomgevingen uit te zetten. Dit heb ik in samenspraak gedaan met de opdrachtgever. Het uitzetten van deze logging zorgt ook voor minder gebruik van het flashgeheugen.

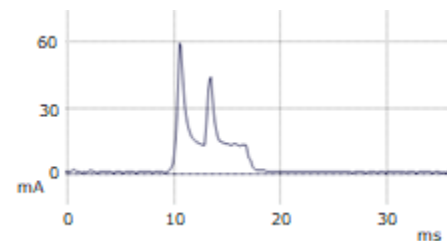


» Afbeelding 8.3, Energieverbruik zonder logging (links) en met logging (rechts). Bovenste lijn: stroomverbruik, Onderste lijn: taak actief (hoog) of inactief (laag)

## ADC: datarate samples

Het uitlezen van de analoge sensor met de ADC gaat met een bepaalde datarate (ook wel samplerate genoemd). In het vorige project is deze op de snelheid 20 SPS gezet. Hierbij kwam de aanbeveling om te onderzoeken of dit verhoogd kan worden. Een hogere samplerate zorgt namelijk voor meer ruis, maar ook zorgt het voor een kortere tijd die de ADC nodig heeft om een meting te doen [4].

In de datasheets van deze externe ADC is te vinden wat de ruisratio's zijn met verschillende datarates en wat de bijbehorende conversietijd is. Hierbij heb ik gekozen voor een gulden middenweg: een datarate van 400 SPS. Deze datarate zorgt namelijk voor een erg beperkte conversietijd (3ms), terwijl voor veel versterkingsfactoren



» Afbeelding 8.4, 400 SPS datarate

de ruis geen of weinig invloed heeft. Aangezien de datarate voorheen op 20 SPS werd gezet is de conversietijd hiermee verlaagd van zo'n 60ms naar 3ms. Dit scheelt in processortijd en de tijd dat de ADC en sensor gevoed moeten worden.

De oppervlakte van de grafiek was in de oude situatie 0,00016 mAh. Nu is de oppervlakte onder de grafiek (afbeelding 8.4) 0,00005 mAh geworden. Dat is gelukt door de datarate te verhogen en door de logging uit te zetten (bespaart 0,00004 mAh oppervlakte). Hiermee wordt ruim voldaan aan requirement 9 (R09).

## **LoRa**

LoRa is de grootste verbruiker van energie op de GEN-node. Dat blijkt uit de stroommetingen die ik heb uitgevoerd. De kleinst gemeten oppervlakte van het verzenden van een bericht was 0,002 mAh en de grootst gemeten oppervlakte was 0,011 mAh.

Zoals te zien in bijlage Ontwerprapport heb ik onderzocht welke LoRa-configuraties er mogelijk zijn en wat de effecten zijn. Uit de testen bleek dat de oppervlakte van het stroomverbruik voor 1 bericht verzenden bestaat uit  $15 \text{ mA} \times \text{airtime}$ . Hierbij is het zendvermogen (TxPower) 16 dBm. Wordt het zendvermogen op bijvoorbeeld 4 dBm gezet daalt de stroom van 15 mA naar zo'n 9 mA. Ik heb dit getest met zowel een KPN-gateway (vanuit huis) als met een Technolution-gateway (op kantoor).

Ik heb onderzoek gedaan naar de airtime omdat deze grote invloed heeft op de oppervlakte van het stroomverbruik. De airtime is afhankelijk van de byte-lengte van het bericht, spreading factor (SF) en de bandbreedte. Omdat er gebruik wordt gemaakt van Adaptive Data Rate (ADR) wordt de spreading factor en bandbreedte automatisch bepaald. In Europa is de bandbreedte bij bijna alle SF-waardes 125 kHz. [17]

Aangezien er geldt dat hoe hoger de SF-waarde is, hoe langer de airtime is [18], wordt er dus een langere tijd minder stroom verbruikt bij een vermindering van de TxPower. Wanneer de SF-waarde met 1 verhoogd wordt betekent dit een verdubbeling voor de airtime en kan zo bij een slecht bereik veranderen van SF7 naar SF12. Dat is vergroting van 32x (25). Dat weegt niet op tegen een iets lager stroomniveau. Het beste is dus om een middenweg te vinden voor TxPower en SF-waarde, waarbij de SF-waarde vaak zo laag mogelijk kan zijn (SF7). SF7 lukt regelmatig met een TxPower van 16 dBm. Daarom is de keuze om de TxPower op 16 dBm te laten.

Metingen samenvoegen om in 1x te versturen scheelt wel in energieverbruik. Een 2x zo lang bericht (byte-length) betekent een airtime die zo'n 1,3x zo groot is [17]. Samenvoegen van 2 berichten bespaart dus zo'n 35% airtime. Hiermee valt winst te halen, maar dat betekent wel dat metingen meer vertraagd worden voordat ze zichtbaar worden in de backoffice. Dit is niet

voor elke gebruiker wenselijk. Daarom is ervoor gekozen met de opdrachtgever hier niet voor te kiezen. Mocht het toch wenselijk zijn bij een specifieke gebruiker, kan de variabele `_nrOfSamplesBeforeSending` worden aangepast in de software.

### Drivers: SPI

Bij het onderzoeken van de SPI-driver merkte ik dat de driver bestaat uit te veel functies met hetzelfde doel. Dit ontwerp heeft als doel deze driver eenvoudiger en kleiner in omvang te maken.

Ten eerste kan de snelheid omhoog van de SPI-communicatie. Dit kan door de baudrate-prescaler te veranderen van 16 naar 8 (van 2 MHz naar 4MHz). De communicatie met deze nieuwe snelheid is gecontroleerd en het levert geen lees- of schrijffouten op.

Ten tweede kunnen functies `SPI_Transfer`, `SPI_Transfer_two_bytes` geschrapt worden omdat deze veel vergelijkingen hebben met `SPI_Transmit`. `SPI_Send` en `SPI_Read` zijn nooit gebruikt en is dus ook overbodig. `SPI_Transmit_Byte` wordt aangepast zodat deze alleen `SPI_Transmit` aanroept. Qua algoritme waren `SPI_Transmit_Byte` en `SPI_Transmit` namelijk hetzelfde, maar hoeft er bij de eerste functie geen data-array te worden geïnitieerd. De nieuwe driver is uitgebeeld in afbeelding 8.5.

SPI
<div>+ SPI_Init() : void + SPI_Transmit(data : uint8 *, len : uint32, receive_buf : uint8 *) : TTISStatus + SPI_Transmit_Byte(data : uint8, receive_buf : uint8 *) : TTISStatus + SPI_DeInitialize() : void</div>

» Afbeelding 8.5, Nieuw ontwerp SPI-driver

### Drivers: I<sup>2</sup>C

De I<sup>2</sup>C-driver bevat voornamelijk functies die grotendeels hetzelfde algoritme uitvoeren. Om dit op te lossen wordt de functie `Hall2cSendReceive` aangepast, deze zal de functies `Hall2cSendDataBuffer` en `Hall2cRecvData` uitvoeren. `Hall2cSendData` is de I<sup>2</sup>C-equivalent van `SPI_Transfer_two_bytes` en roept alleen `Hall2cSendDataBuffer` aan. `Hall2cSendDataBuffer` en `Hall2cRecvData` zullen de gedeeltes met hetzelfde algoritme uitvoeren in de nieuwe functie `Hall2cSendAddress`. Het aangepaste driverontwerp wordt getoond in afbeelding 8.7.

I2C
<ul style="list-style-type: none"> <li>- Hall2CSendAddress(address: uint8) : TTISStatus</li> <li>+ Hall2cInit(clockSpeed : uint32) : void</li> <li>+ Hall2cSendData(address : uint8, data : uint8) : TTISStatus</li> <li>+ Hall2cSendDataBuffer(address : uint8, data : uint8*, len : uint8) : TTISStatus</li> <li>+ Hall2cSendReceive(address : uint8, txData : uint8*, txLen : uint8, rxData : uint8*, rxLen : uint8) : TTISStatus</li> <li>+ Hall2cRecvData(address : uint8, data : uint8*, nbrOfBytes : uint8) : TTISStatus</li> <li>+ ResetStatesI2c() : void</li> <li>+ Hall2cDeinit() : void</li> </ul>

» Afbeelding 8.6, Nieuw ontwerp I2C-driver

## 8.2 Ontwikkefase

Oriëntatie- fase	Analyse- fase	Iteratie 1	Iteratie 2	Iteratie 3	Advies- fase
		Ontwerpfase	Ontwikkefase	Testfase	

### 8.2.1 Energie

In hoofdstuk 6 was al naar voren gekomen dat de informatieve logging geschrapt moet worden in verband met het beperkte flash-geheugen bij de firmwareapplicatie multi-sensor. In hoofdstuk 8.1 is naar voren gekomen dat de informatieve logging altijd uit moet staan in elke firmwareapplicatie die bestemd is voor een productieomgeving. Daarom heb ik het volgende uitgevoerd.

Door de waarde van de define `TL_LOG_LEVEL_THRESHOLD` aan te passen naar `TL_LOG_LEVEL_NOTICE` in `tl_log.h` of toe te voegen in de makefile (script dat de compiler aanstuurt) worden de informatieve logberichten (`TLogInfo`, `TLogDebug`) verwijderd uit de code door de precompiler. Wanneer er debugging moet plaatsvinden wordt de waarde van de define veranderd naar `TL_LOG_LEVEL_INFO` of `TL_LOG_LEVEL_DEBUG`. Op deze manier kan er snel geschakeld worden tussen het compileren van een firmwarebestand met of zonder informatieve logging. Het levert een besparing op van 6kB (6032 bytes) voor de applicatie multi\_sensor.

Daarnaast heb ik de datarate van de ADC en bijbehorende wachttijden aangepast in de functies `AdcExternalInit` en `AdcExternalReadData` in de module van de externe ADC: `adc_ADS114S06B`.

### 8.2.2 Drivers

De drivers heb ik aangepast volgens het ontwerp in hoofdstuk 8.1. Allereerst heb ik de SPI-driver aangepast. Dit heeft geleverd tot een besparing van 280 bytes aan flash-geheugen.



Daarna heb ik de I2C-driver aangepast volgens het ontwerp. Dit levert voor deze driver een besparing van 324 bytes aan flash-geheugen op. De functies bevatten nu minder dubbelzinnige code waardoor de onderhoudbaarheid verbetert.

## 8.3 Testfase

Oriëntatie-fase	Analyse-fase	Iteratie 1	Iteratie 2	Iteratie 3	Advies-fase
		Ontwerpfase	Ontwikkelfase	Testfase	

In de testfase wordt de werking van de drivers gevalideerd. Ook wordt gecontroleerd of het energieverbruik bij het doen van een meting met de externe ADC lager is geworden. Dat gebeurt op basis van de twee requirements (R01 en R09) die deze iteratie de focus hadden.

**Beschrijving:** De firmware heeft driverondersteuning voor de ingangen Analog In, SPI en I2C en de uitgangen Analog Out en Digital Out.

**R01**

**Precondities:** UART wordt uitgelezen

### Acties:

- Sluit een ondersteunde I2C-sensor aan en stel DIP-switch in.
- Zet de GEN-node aan
- Bekijk de meting
- Sluit een PT500 of soortgelijke sensor aan en zet de DIP-switch op 1001
- Reset de GEN-node met de magneet
- Bekijk de meting

### Verwacht resultaat:

Console geeft aan dat het uitlezen van de I2C-sensor is voltooid door een meetwaarde te tonen (*I2C en Digital Out*). Het uitlezen van een PT500 is gelukt door het tonen van een acceptabele meetwaarde: tussen 0,1 en 2,4 V (*SPI, Analog In en Analog Out*).

### Daadwerkelijk resultaat:

De sensoren geven acceptabele meetwaardes waarmee de driverondersteuning is aangetoond.

**OK** **Ja**

**Beschrijving:** Het energieverbruik is lager dan het huidig verbruik wanneer de GEN-node wordt toegepast voor het uitlezen van analoge sensoren met de externe ADC.

**R09**

**Precondities:** Sluit binnen de stroomkring een oscilloscoop aan. Configureer de opstelling:

- Probe 1 binnen de stroomkring aansluiten met kleine weerstand (ordegrootte 200 mV)
- Probe 2 aansluiten op 3v3\_switched (ordegrootte 2V)
- Tijdvakken op 10 ms
- Trigger op omhooggaande flank van probe 2

**Acties:**

- Plaats de nieuwe firmware op de GEN-node
- Zet de GEN-node aan op DIP-stand 0000 (analoge sensor, gain 1)
- Wacht totdat de ADC voor de tweede keer heeft gemeten
- Bekijk de meting van de oscilloscoop
- Reken de spanningen van probe 1 om naar stroom (wet van Ohm)

**Verwacht resultaat:**

De oppervlakte onder de grafiek is kleiner dan 0,00016 mAh.

**Daadwerkelijk resultaat:**

De oppervlakte is 0,00005 mAh.

**OK** **Ja**

## **Conclusie**

Met deze tests is aangetoond dat de aanpassingen die aan de I2C- en SPI-drivers zijn gedaan ervoor gezorgd hebben dat beiden protocollen nog steeds werken op de GEN-node. Ook is het energieverbruik gedaald tijdens het uitlezen van de ADC. De opdrachtgever was tevreden met dit resultaat.

## 9. Adviesfase

Oriëntatie-fase	Analyse-fase	Iteratie 1	Iteratie 2	Iteratie 3	Advies-fase
-----------------	--------------	------------	------------	------------	-------------

De laatste fase is de adviesfase. Tijdens de adviesfase worden de eerst conclusies opgesteld. Vervolgens worden de aanbevelingen geformuleerd en daarna wordt de benodigde overdracht beschreven. Dit geheel wordt verwerkt in het adviesrapport.

De conclusies heb ik opgesteld door te kijken in of de deelvragen uit de probleemstelling zijn beantwoord. Wanneer de requirements in voldoende mate voldaan zijn wordt er beoordeeld is het doel behaald is. Dit is beschreven in hoofdstuk 10.

De aanbevelingen voor het gebruik van de firmware heb ik hieronder opgesteld.

### **Grotere LoRa-module**

Om in de toekomst nieuwe sensoren te kunnen toevoegen aan de multi\_sensor-applicatie, of om de eRPC-functionaliteit bij deze applicatie weer te kunnen gebruiken raad ik aan over te stappen naar een andere LoRa-module met een groter flash-geheugen. De fabrikant van de huidige LoRa-module raadt de WiMod iM881A-XL aan. Deze wordt verwacht in het tweede of derde kwartaal van 2020. Deze heeft 192 kB flash. Onbekend is nog hoe groot de statische bibliotheek is.

### **Metingen opsparen**

Metingen opsparen zorgt voor een kortere totale airtime (35% besparing bij 2 berichten samenvoegen) en daardoor een lager energieverbruik. Door de waarde van `_nrOfSamplesBeforeSending` te veranderen kan dit worden ingeschakeld. Omdat elke situatie anders is, staat deze waarde standaard op 1. Afhankelijk van de requirements van de klant zou deze waarde vergroot kunnen worden.

### **Logging uit voor productieomgeving**

Wanneer debugging niet nodig is, raad ik aan om in ieder geval de informatieve logging uit te zetten. Dit bespaart energie en bespaart flash-geheugen. Voor debugging op locatie is het handig om alvast twee versies van de firmware-applicatie te hebben: één met logging en één zonder logging. Voor de applicatie multi\_sensor is dit lastiger, aangezien het inschakelen van logging niet past in het 128 kB flash-geheugen. Maak daarom gebruik van twee debugversies, met elk een gedeelte van de ondersteunde sensoren.

### **Backoffice omrekeningen**

Metingen van analoge sensoren worden nu verstuurd naar de backoffice in millivolts. Voor een grotere gebruiksvriendelijkheid kan hier in de backoffice geprogrammeerd worden dat er voor een GEN-node een omrekenformule kan worden opgegeven. Dat vertaalt de waardes in millivolt naar een nuttige eenheid die past bij de aangesloten sensor.

### **Kalibratie Alphasense-sensoren**

De fabriekskalibratiewaarden bleken bij testen onvoldoende. Regelmatig herkalibreren is nodig om nuttige informatie uit deze sensoren te achterhalen. De nieuwe kalibratiewaardes kunnen vervolgens geplaatst worden in de omrekenformule die bij een specifieke GEN-node wordt opgeslagen in de backoffice.

Tot slot heb ik een overdracht opgesteld waarin de software en documentatie zijn gearchiveerd door middel van SVN (Software Version Control) in een bestaande repository.

# 10. Conclusie

De hoofdvraag van dit onderzoek was “Hoe kan de firmware van de GEN-node verbeterd worden?”. De beantwoording van deze probleemstelling is opgesplitst in drie deelvragen.

## **Hoe kunnen analoge sensoren universeel worden ondersteund met sensorkeuze?**

Door middel van de firmware dat ontworpen en ontwikkeld is in hoofdstuk 6.1 en 6.2 kunnen de gewenste analoge sensoren uitgelezen worden op de GEN-node. Dat is gedaan door aansturing te ontwikkelen voor de externe ADC waarvan de aansturing een groot aantal analoge sensoren ondersteund zoals een 4-20 mA-sensor en de 0-2,5 volt-sensor. Met de DIP-switch is op de GEN-node een keuze te maken in het gewenste lijstje sensoren.

## **Hoe kunnen geselecteerde sensoren ten behoeve van het meten van luchtkwaliteit worden ondersteund?**

Hierbij is een specifieke aansturing voor de geselecteerde Alphasense-sensoren ontworpen en ontwikkeld (zie hoofdstuk 7.1 en 7.2). Het grootste verschil met de voorgaande deelvraag is dat deze sensoren twee elektroden bevatten die los van elkaar gemeten moeten worden. Dit is nodig voor een juiste omzetting van millivolts naar parts-per-billion.

## **Hoe kan het energieverbruik worden verlaagd?**

Het energieverbruik is verlaagd door de informatieve logging uit te zetten. Voor het uitlezen van een analoge sensor is het energieverbruik extra verlaagd door het hoger zetten van de datarate van de ADC. Het energieverbruik zou verder nog verlaagd kunnen worden wanneer metingen opgespaard worden voordat ze verzonden worden.

Daarnaast zijn er ook requirements getest (hoofdstuk 6.3, 7.3 en 8.3). Het resultaat hiervan is dat aan de requirements behorende bij iteratie 1 en 2 en 3 is voldaan. Wel met de kanttekening dat requirement R04 (versterking) qua softwareaansturing werkt, maar dat de beschikbare hardware voor de test niet het verwachte resultaat geeft. Hiervoor komt nieuwe hardware beschikbaar. Aan requirement R06 (kunnen schakelen tussen sensoren) is ook voldaan, maar het is momenteel alleen mogelijk wanneer er enkele functionaliteiten komen te vervallen: eRPC en informatieve logging. Requirement R08 (generieke ondersteuning van I<sup>2</sup>C- en SPI-sensoren), is een could-have en is niet aan bod gekomen. Dit zou te veel tijd kosten. De niet-functionele requirements R10, R11 en R12 hebben geen test, maar er is wel rekening mee gehouden tijdens het proces.

Het doel “De afstudeeropdracht leidt tot verbeterde firmware voor een Proof-of-Concept-model GEN-node die ondersteuning heeft voor analoge sensoren, plus geselecteerde

sensoren voor het meten van luchtkwaliteit waarbij het energieverbruik verlaagd is.” is hiermee behaald.

Het advies is om zodra deze beschikbaar is de iM881A-XL LoRa-module toe te passen omdat deze meer flashgeheugen heeft. Daarmee is het mogelijk zonder concessies te voldoen aan requirement R06.

# 11. Evaluatie

## 11.1 Kwaliteit van producten

Tijdens dit project heb ik een zestal (tussen)producten moeten opleveren. Het gaat om 5 rapporten en de firmware. De genoemde werkzaamheden in hoofdstuk 3.6 geven de activiteiten aan voor deze producten. Het Plan van Aanpak is meer procesgericht en wordt daarom besproken in hoofdstuk 11.2.

Het analyserapport bevat een analyse van de huidige situatie en de gewenste situatie. De gewenste situatie richt zich vooral op de mogelijkheden van de wensen. Hierdoor is het opstellen van de requirements makkelijker geworden. Echter mist er voor een analyse aanvullend onderzoek voor de tweede en derde iteratie. Dit onderzoek komt daardoor pas in het ontwerprapport aan bod.

Het ontwerprapport heeft op zijn beurt meer onderzoek dan gewenst voor een ontwerprapport. Het ontwerprapport bevat diagrammen op eenvoudig en gedetailleerd niveau. Dat is goed, aangezien de lezer hiermee niet te snel in het diepe wordt gegooid. Nadeel is dat voor de hoofdstukken met de 2<sup>de</sup> en 3<sup>de</sup> iteratie nauwelijks diagrammen hoefden te worden toegevoegd of gewijzigd. Dit maakt dat het hoofdstuk in het ontwerprapport over de 1<sup>e</sup> iteratie een flinke kluif is aan informatie in vergelijking met de andere hoofdstukken.

De firmware is ontwikkeld en voldoet naar mijn mening aan voldoet aan de requirements van de drie iteraties. Wel had ik graag gezien dat de firmware gefunctioneerd had zonder dat hierbij de eRPC geschrapt moesten worden en/of dat er een versie met informatieve logging was die binnen het flash-geheugen past met alle ondersteunde sensoren erin voor debugging (beschreven in hoofdstuk 6.2.2).

Voor het testrapport was het de bedoeling om veldtesten uit te voeren. Het testrapport bevat een testplan volgens de requirements die bij de iteraties horen. De testen in het testrapport zijn uitgevoerd volgens de opgestelde testprocedure. Dit is naar mijn mening volgens plan verlopen.

In het adviesrapport zijn de conclusies van het project opgesteld. Dit is gedaan op basis van de requirements, de probleemstelling en doelstelling. Vervolgens zijn de adviezen onderbouwt en is er een overdracht opgenomen. Voor het adviesrapport was bepaald in het afstudeerplan dat hierin de conclusies en adviezen onderbouwt zijn, wat voldoende is gebeurd.

## 11.2 Procesevaluatie

In het Plan van Aanpak is er nagedacht hoe het proces ingedeeld gaat worden. In deze paragraaf wordt het proces geëvalueerd en wordt bekeken in hoeverre dit overeenkomt met wat er bedacht is in het Plan van Aanpak.

### **Oriëntatie (Plan van Aanpak)**

In het Plan van Aanpak heb ik een planning gemaakt volgens de gestelde fasering (hoofdstuk 4.3.2, tabel 4.7). Deze planning was vrij ruim merkte ik tijdens de uitvoering van het project. Het was lastig vertellen wat de voortgang was in bijvoorbeeld een blok van 4 weken gereserveerd voor de analysefase. Het had beter geweest als er een gedetailleerde planning is opgenomen met een indeling van werkzaamheden die bij de fase horen.

### **Analyse**

In het afstudeerplan bevatte het begin van de analyse de volgende werkzaamheden: Eerst zou er literatuur bestudeerd gaan worden en vervolgens wordt de huidige situatie geanalyseerd. Ik had besloten dit anders in te delen tijdens het uitvoeren van de analysefase. Dat had er mee te maken dat er sprake was van ontbrekende kennis over de GEN-node en Sense2Grow. Daarom begon ik met het analyseren van de huidige situatie en literatuuronderzoek voor de onderdelen waarvan ik nog niet genoeg wist. Daarna was ik van plan om (literatuur)onderzoek te starten naar de onderdelen die gewenst zijn voor de nieuwe GEN-node. Ik denk dat dit een goede zet is geweest aangezien ik nu doelgerichter te werk kon gaan met de literatuurstudie. Ook kon ik met het onderzoeken van de gewenste GEN-node de requirements gemakkelijker opstellen.

### **Iteratie 1: ondersteunen analoge sensoren en de multi-sensor-applicatie**

Bij aanvang van de 1<sup>e</sup> iteratie merkte ik dat de omvang nogal groot is in vergelijking met de andere 2 iteraties. Dat merkte ik omdat er in deze iteratie twee applicaties ontwikkeld moesten worden. Ik heb er toen voor gekozen het ontwerp en ontwikkeling in tweeën te delen: een gedeelte analoge sensor en een gedeelte multi-sensor. Dit had ik mede gedaan met de reden dat de multi-sensorgedeelte het risico bracht niet mogelijk te zijn vanwege een beperkt flash-geheugen. Dit risico werd werkelijkheid en heb ik vervolgens volgens de risicoanalyse opgelost. De uitwerking hiervan stemde de opdrachtgever tevreden. Achteraf gezien was het geen slecht idee geweest als ik van deze iteratie twee aparte iteraties had gemaakt. Dan zou de omvang van de opdracht beter verdeeld zijn over de iteraties. Desalniettemin heeft een goed onderzoek naar de wensen voor deze iteratie in de analysefase, samen met de risicoanalyse ervoor gezorgd dat deze iteratie binnen de vier geplande weken af was.



## **Iteratie 2: Luchtkwaliteit**

De tweede iteratie zou zich net als de eerste iteratie richten op analoge sensoren. Voor het meten van luchtkwaliteit viel het oog op de twee Alphasense-sensoren. Dit zijn de driedraadssensoren. Omdat in de analyse een ander soort driedraadssensor naar voren kwam (de RTD PT500) leek dit een goede combinatie te zijn. Ik had gedacht dat de meetmethode van de Alphasense- en de RTD-sensoren vrijwel gelijk zouden zijn. Bij de uitvoering van de iteratie bleek dat de twee Alphasense-sensoren niet in 1 meting gecombineerd kon worden, wat wel het geval is bij de RTD. Hier kwam ik achter toen ik de gemeten spanning van de Alphasense NO2-B43F wilde controleren door dit te vertalen naar ppb middels de opgenomen grafiek in de datasheet van deze sensor. De waardes leken niet te kloppen. Het kostte veel tijd om achter de juiste documentatie te komen van deze sensoren met omrekenformules en kalibratie-informatie. Hier kwam ik erachter dat voor een goede omrekening van volt naar ppb de twee uitgangsspanningen los van elkaar moeten worden gemeten. Terugkijkend had ik deze informatie al in de analysefase moeten hebben. Gelukkig waren de drie weken die voor deze iteratie ingepland waren voldoende om de Alphasense-sensoren en de RTD te ondersteunen.

## **Iteratie 3: Energie**

Tijdens het uitvoeren van iteratie 3 merkte ik dat de verdeling van ontwerpen, ontwikkelen en testen niet geheel passen bij de uit te voeren werkzaamheden tijdens deze iteratie. In het Plan van Aanpak was verwacht dat er grote stukken code herschreven moesten worden om ervoor te zorgen dat het systeem minder vaak terug hoeft te gaan naar het OS. Tijdens de uitvoering van de iteratie heb ik gecontroleerd of dit ook moest en dit bleek niet het geval te zijn. In werkelijkheid bestonden de werkzaamheden daardoor veel meer uit het onderzoeken van het energieverbruik en daarop wijzigingen aanbrengen. Er wordt onderzocht hoe het energieverbruik verbeterd kan worden. Maar omdat het precieze resultaat onbekend is, is vooraf niet duidelijk of dit uiteindelijk mogelijk is, in hoeverre dit mogelijk is en op welke manieren dit mogelijk is. Het te verwachten resultaat is daarom niet heel afgebakend, iets dat niet goed past bij de RAD-methode. Een projectmethode dat meer gericht is op veranderende inzichten en omstandigheden (zoals scrum) zou beter passen bij deze iteratie. Deze iteratie is ook binnen de geplande 3 weken voltooid.

## **Thuiswerken**

Wegens de maatregelen van het coronavirus werd ik genoodzaakt zoveel mogelijk thuis te werken. Vanaf de ontwerpfase van de eerste iteratie heb ik thuisgewerkt. Gelukkig was het mogelijk om naar kantoor te komen om sensoren te halen en bijvoorbeeld de oscilloscoop. Voor de communicatie met mijn begeleider werd het wel iets lastiger. Even snel meekijken of even sparren werd al wat lastiger. Toch heb ik geprobeerd regelmatig het contact te

behouden middels Google Hangouts. De wekelijkse meeting met mijn begeleider en opdrachtgever werd een videogesprek. Dit werkt best goed, met als enig nadeel dat je de GEN-node niet fysiek kan laten testen door mijn opdrachtgever. Voor het gebruikmaken en testen van de LoRa-verbinding thuis had ik bedacht gebruik te maken van een KPN-gateway. Mijn begeleider heeft vervolgens een koppeling gemaakt zodat ik de berichten van de KPN-gateway kon laten doorsturen naar de backoffice van Technolution.

### **Doel en probleem**

Het doel is: “De afstudeeropdracht leidt tot verbeterde firmware voor een Proof-of-Concept-model GEN-node die ondersteuning heeft voor analoge sensoren, plus geselecteerde sensoren voor het meten van luchtkwaliteit waarbij het energieverbruik verlaagd is”. Met het beschreven proces is het gelukt om het doel te bereiken.

Met het uitvoeren van de drie iteraties is er antwoord gegeven aan de drie deelvragen van de probleemstelling “Hoe kan de firmware van de GEN-node verbeterd worden?”. Hiermee is het probleem opgelost.

## 12. Aantonen beroepstaken

Voor het afstuderen is het belangrijk dat er aan enkele beroepstaken voldaan is tijdens dit project. De beroepstaken zijn vastgesteld in het afstudeerplan, waarbij beroepstaak A1, Gc en Gf verplicht zijn voor de afstudeerrichting IoT. Dit hoofdstuk beschrijft waar ik aantoon op welke wijze de beroepstaken voldaan zijn.

### **A1 Analyseren probleemdomein & opstellen probleemstelling.**

Toon ik aan in het Plan van aanpak en het Analyserapport.

Het analyseren van het probleemdomein is gebeurd in het Plan van Aanpak (en in hoofdstuk 2.2) waarbij het Sense2Grow-netwerk het probleemdomein is waarin de GEN-node zich bevindt. Dit is verder geanalyseerd in het Analyserapport (en in hoofdstuk 5.1) onder het hoofdstuk Huidige GEN-node. In het Plan van Aanpak is de probleemstelling opgesteld. In hoofdstuk 4.1 van het afstudeerdossier is het proces naar de huidige probleemstelling beschreven.

### **C10 Ontwerpen embedded & realtimes systemen**

Toon ik aan in het Ontwerprapport.

Het ontwerpen van het embedded systeem is te vinden in het Ontwerprapport. Het proces naar dit rapport is beschreven in de hoofdstukken 6.1, 7.1 en 8.1 van het afstudeerdossier. Dit leverde een ontwerp op voor het toepassen van diverse sensoren dat voortbouwt op een bestaand ontwerp.

### **D14 Realiseren van software**

Toon ik aan met de firmware Proof of Concept.

Met de firmware laat ik zien dat het ontwerp is gerealiseerd. In de hoofdstukken 6.2, 7.2 en 8.2 van dit afstudeerdossier zijn enkele fragmenten te vinden van de code. De software bouwt voort op al bestaande software.

### **D15 Testen**

Toon ik aan in het Testrapport.

In het Testrapport zijn testprocedures geschreven waarbij de requirements getest worden. De testen zijn vervolgens uitgevoerd. In hoofdstuk 6.3 van het afstudeerdossier is één test volledig te zien. De overige testen zijn op soortgelijke wijze aangepakt en zijn in te zien in het Testrapport.

### **Gc Kritisch, onderzoekend en methodisch werken**

Toon ik aan in het afstudeerdossier in de hoofdstukken waarin het verlopen proces wordt beschreven.

In het afstudeerdossier is in hoofdstuk 8.1 beschreven dat er kritisch is gekeken naar het terugkeren naar het OS, waarbij er processortijd en dus energie bespaard zou kunnen worden door de software te herschrijven. Dit bleek niet zo te zijn. Ook wordt er

onderzoekend gewerkt. Zo wordt er in hoofdstuk 7.2 beschreven dat er problemen zijn met het uitlezen van de Alphasense-sensoren. Aangezien dit zich meer in het elektronica domein bevond moest ik onderzoeken hoe deze sensoren precies werken en reageren. Tot slot wordt er een aangepaste variant van de RAD-methode gevolgd. Dit is terug te zien in de hoofdstukken die ingedeeld zijn volgens de fasering zoals dit was vastgesteld in hoofdstuk 4.3.

**Gf      Leren leren: voorbereiden op volgende studiefase en beroep**

Toon ik aan in het hoofdstuk evaluatie binnen het afstudeerdossier.

In hoofdstuk 11 is het project geëvalueerd. Hieruit blijkt dat de gekozen methode wat minder goed van kracht komt wanneer er sprake is van een niet geheel afgebakende uitkomst, dat bij iteratie 3 (energieverbruik) het geval is.

## 13. Verklarende woordenlijst

**AE** Auxiliary Electrode, hulpelektrode. Reageert niet op de sensor maar wel op omgevingsfactoren.

**Gain** De versterkingsfactor, is in te stellen op de PGA.

**LoRa** Bevindt zich in de datalinklaag van het OSI-model. Dit protocol zorgt voor draadloze communicatie.

**LoRaWAN** Bevindt zich in de datalinklaag (LoRa) en de netwerklaag van het OSI-model. Binnen LoRaWAN is er de koppeling naar het internet (cloud) vanuit de gateway.

**Nulafwijking** Zero error. Staat voor de afwijking vanaf nul op het moment dat de gemeten waarde nul zou moeten zijn.

**Oscilloscoop** Meetinstrument. Kan spanningen meten over een kort tijdsbestek.

**PGA** Dit is een versterker dat ingebouwd is in de ADC. Deze versterker is aan te sturen door middel van SPI-commando's.

**RTD** Temperatuurssensor. Bestaat uit een weerstand die gevoelig is voor temperatuur. Voorbeelden zijn de PT100, PT500 en PT1000. Het getal na PT is de weerstandswaarde in Ohm bij 0 graden Celcius.

**Sense2Grow** Platform van Technolution voor sensornetwerken, waarbinnen de GEN-node werkzaam is.

**SPI** Serial Peripheral Interface is een datalink tussen bijvoorbeeld de microcontroller en de ADC. Hierdoor is tussen deze componenten communicatie mogelijk.

**UART** Universal asynchronous receiver-transmitter, hiermee kan de logging van de node worden ingezien aan de hand van een seriële communicatie.

**WE** Working Electrode, werkzame elektrode dat reageert op de sensor.

**Wet van Ohm** Natuurkundewet, geeft het verband weer tussen spanning (U), stroom (I) en weerstand (R). Notatie:  $U = I \times R$

# Literatuur

- [1] N. Labordus, „Design Sense2Grow GEN node,” Technolution, Gouda, 2020.
- [2] IMST GmbH, „Datasheet WiMOD iM880B,” Kamp-Lintfort, 2016.
- [3] M. d. Rooij, „Internship Report GEN,” Gouda, 2018.
- [4] Texas Instruments, „Datasheet ADS114S0xB,” 2017.
- [5] Sparkfun, „Analog to Digital Conversion: Relating ADC Value to Voltage,” [Online]. Available: <https://learn.sparkfun.com/tutorials/analog-to-digital-conversion/relating-adc-value-to-voltage>. [Geopend 19 februari 2020].
- [6] Technolution, „Sense and Grow - Basic design node software,” Gouda, 2016.
- [7] Alphasense, „Datasheet Individual Sensor Board (ISB) Alphasense B4 4-Electrode Gas Sensors,” [Online]. Available: <http://www.alphasense.com/WEB1213/wp-content/uploads/2019/10/ISB.pdf>. [Geopend 6 april 2020].
- [8] S. V. K. Krishnaswamy, Industrial Instrumentation, Delhi: New Age International, 2005.
- [9] Alphasense, „Datasheet NO2-B43F Nitrogen Dioxide Sensor,” [Online]. Available: <http://www.alphasense.com/WEB1213/wp-content/uploads/2019/09/NO2-B43F.pdf>. [Geopend 10 maart 2020].
- [10] Alphasense, „Datasheet SO2-B4 Sulfur Dioxide Sensor,” [Online]. Available: <http://www.alphasense.com/WEB1213/wp-content/uploads/2019/09/SO2-B4.pdf>.
- [11] STMicroelectronics, „Reference Manual STM32L151C6-a,” 2020.
- [12] Alphasense, „AAN 803-01 - Correcting for background currents in four electrode toxic gas sensors,” mei 2014. [Online]. Available: <https://zueriluft.ch/makezurich/AAN803.pdf>. [Geopend 16 mei 2020].
- [13] RIVM, „Notitie Kalibratie van NO2 sensoren,” 10 januari 2018. [Online]. Available: <https://www.samenmetenaanluchtkwaliteit.nl/documenten/kalibratie-van-no2->

sensoren. [Geopend 14 mei 2020].

- [14] A. Gonzales, A. Boies, J. Swason en D. Kittelson, „Field Calibration of Low-Cost Air Pollution Sensors,” 19 augustus 2019. [Online]. Available: <https://www.atmos-meas-tech-discuss.net/amt-2019-299/amt-2019-299.pdf>. [Geopend 17 mei 2020].
- [15] B. Mijling, Q. Jiang, D. d. Jonge en S. Bocconi, „Field calibration of electrochemical NO<sub>2</sub> sensors in a citizen,” 5 maart 2018. [Online]. Available: <https://www.atmos-meas-tech.net/11/1297/2018/amt-11-1297-2018.pdf>. [Geopend 17 mei 2020].
- [16] J. Wesseling, D. Drukker, P. Zandveld en L. Gast, „Meten van NO<sub>2</sub> met goedkope sensoren,” 18 september 2017. [Online]. Available: [https://www.samenmetenaanluchtkwaliteit.nl/sites/default/files/2017-09/RIVM%20notitie%20meten%20van%20NO<sub>2</sub>%20met%20goedkope%20sensoren.pdf](https://www.samenmetenaanluchtkwaliteit.nl/sites/default/files/2017-09/RIVM%20notitie%20meten%20van%20NO2%20met%20goedkope%20sensoren.pdf). [Geopend 14 mei 2020].
- [17] The Things Network, „Airtime calculator,” [Online]. Available: <https://www.thethingsnetwork.org/airtime-calculator>. [Geopend 8 mei 2020].
- [18] E. Bäumker, „Minimizing power consumption of LoRa® and LoRaWAN for low-power wireless sensor nodes,” 2019. [Online]. Available: <https://iopscience.iop.org/article/10.1088/1742-6596/1407/1/012092/pdf>.

Bijlage A

## **Plan van Aanpak**



# Verbeterde firmware GEN-node



## Plan van Aanpak

*Februari 2020, Gouda*

**Opdrachtgever**

Olaf Peters

**Auteur**

Jeremy Hut



DE HAAGSE  
HOGESCHOOL

## Versiebeheer

<b>Versie</b>	<b>Datum</b>	<b>Opmerkingen</b>	<b>Auteur</b>
1.0	14-02-2020	Initiële versie	Jeremy Hut
1.1	17-02-2020	Feedback van Jordy Huiting verwerkt	Jeremy Hut

# Inhoudsopgave

<b>1. Inleiding</b>	<b>3</b>
<b>2. Technolution</b>	<b>4</b>
2.1 Organisatiestructuur	4
2.2 Probleemdomein	5
<b>3. Opdracht</b>	<b>6</b>
3.1 Aanleiding	6
3.2 Probleemstelling	6
3.3 Doelstelling	6
3.4 Resultaat	6
3.5 Randvoorwaarden en afbakening	6
<b>4. Risicoanalyse</b>	<b>8</b>
<b>5. Aanpak</b>	<b>10</b>
5.1 Methode	10
5.2 Werkzaamheden	10
5.3 Planning	11

# 1. Inleiding

Dit Plan van Aanpak is geschreven voor het afstudeerproject van Jeremy Hut, student HBO-ICT bij De Haagse Hogeschool. De afstudeerstage wordt uitgevoerd in opdracht van Technolution B.V. in Gouda. Technolution is een software- en elektronicabedrijf, wat goed aansluit bij het afstudeerprogramma Internet of Things van de differentiatie Networks en Systems Engineering.

Dit project bestaat uit het verbeteren van de firmware van de GEN-node, een generieke I/O-module met LoRa-communicatie. In de praktijk zal deze node gebruikt worden voor het uitlezen en versturen van sensordata. De GEN-node is een onderdeel van het Sense2Grow-netwerk. Dit netwerk bestaat uit verschillende sensoren, bijvoorbeeld een luchtdrukmeter of een lichtintensiteitsensor. Door middel van communicatie via een LoRa-verbinding door het LoRaWAN-netwerk wordt de sensordata gecommuniceerd en is vervolgens in een web-omgeving inzichtelijk.

In dit document, het Plan van Aanpak, zal de organisatie beschreven worden. Daarnaast wordt het probleem en het doel geformuleerd. Vervolgens is er een risicoanalyse te vinden en tot slot wordt de aanpak beschreven inclusief een planning.

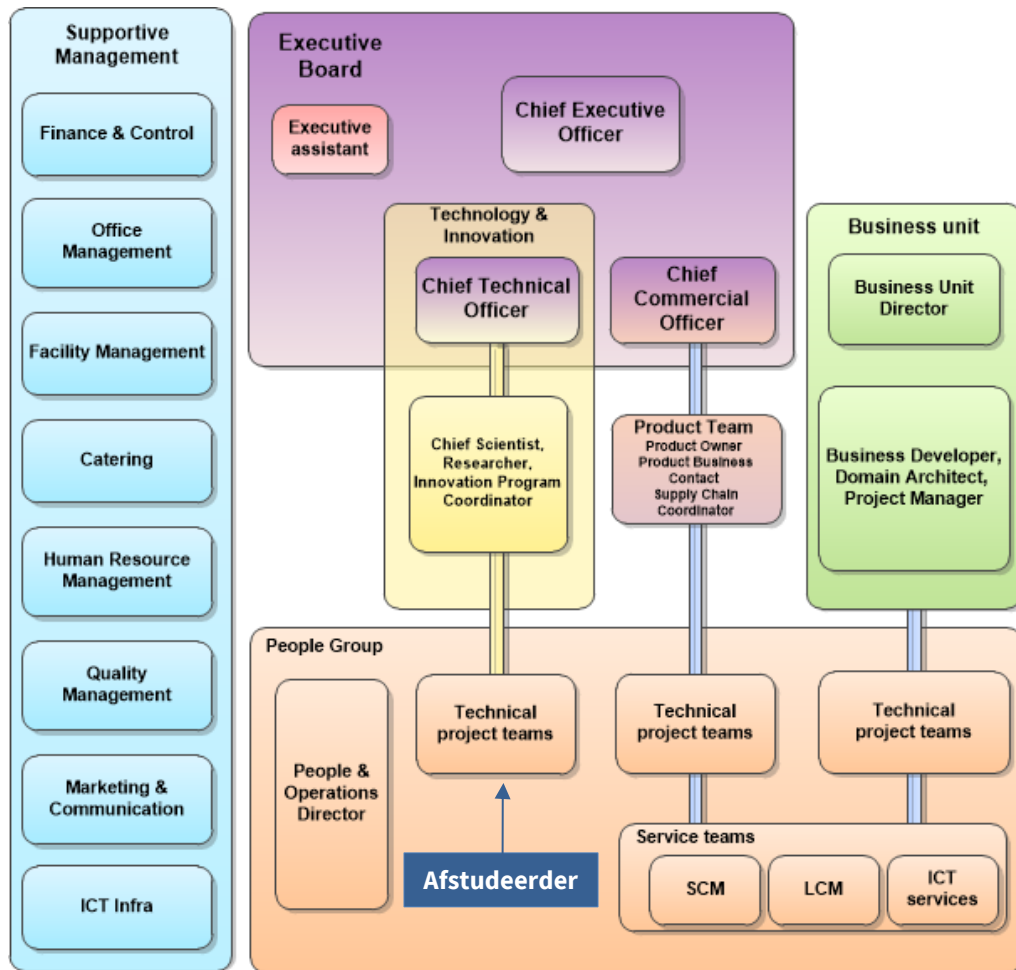
## 2. Technolution

Technolution is een commercieel bedrijf dat zich richt op de ontwikkeling van elektronica, programmeerbare logica en (embedded) software voor diverse organisaties. De missie van het bedrijf is het maken van technische oplossingen die waardevol zijn voor de klant. De hoofdvestiging is in Gouda en er is tevens een vestiging in Deventer. Het afstuderen wordt gedaan in Gouda.

### 2.1 Organisatiestructuur

In de vestiging in Gouda werken meer dan 200 medewerkers. De projecten binnen het bedrijf zijn verdeeld over vier business units: mobiliteit, energie, industrie en Public Safety & Security. Dit project valt onder thema energie. Hier houdt het bedrijf zich bezig met het Internet of Things voor een slim energienetwerk. Al deze projecten zijn te vinden onder de 'technical project teams' in het organigram (afbeelding 2.1). Hier is ook mijn positie in de organisatie te vinden. De technical project teams worden aangestuurd door de hogere lagen in het organigram.

Binnen de organisatie heb ik te maken met de opdrachtgever Olaf Peters en begeleider Jordy Huiting. Olaf Peters heeft binnen het bedrijf de functie Domein Architect en Jordy Huiting is System Designer. Er is sprake van een matrixorganisatie, waarbij er een informele cultuur binnen de projecten wordt nagestreefd. Deze organisatiesoort typeert zich door de indeling in business units en het projectgericht werken, waarbij elk project een eigen team en verantwoordelijken heeft.



» Afbeelding 2.1, organigram Technolution

## 2.2 Probleemdomein

Het probleemdomein van dit project is het overkoepelende project genaamd Sense2Grow. Hier hebben naast de vaste medewerkers ook al een WO- en een HBO-afstudeerder aan gewerkt. Het netwerk wordt door middel van sensoren, gekoppeld via LoRa, informatie verzameld in een dashboard. Eén van de bedrijven die hier gebruik van maakt is Tata Steel.

## 3. Opdracht

### 3.1 Aanleiding

Technolution heeft een bestaand IoT-netwerk Sense2Grow. In dit netwerk hangen LoRa-sensornodes, de GEN-node, die enkele sensoren ondersteund. De ondersteuning van het aantal sensoren is beperkt. Technolution wil een nieuwe generatie GEN-node.

### 3.2 Probleemstelling

De hoofdvraag van dit onderzoek is:

Hoe kan de firmware van de GEN-node verbeterd worden?

Om de hoofdvraag te beantwoorden dienen de volgende deelvragen te worden beantwoord:

- Hoe kunnen analoge sensoren universeel worden ondersteund met sensorkeuze?
- Hoe kunnen geselecteerde sensoren ten behoeve van het meten van luchtkwaliteit worden ondersteund?
- Hoe kan het energieverbruik worden verlaagd?

### 3.3 Doelstelling

De afstudeeropdracht leidt tot verbeterde firmware voor een Proof-of-Concept-model GEN-node die ondersteuning heeft voor analoge sensoren, plus geselecteerde sensoren voor het meten van luchtkwaliteit waarbij het energieverbruik verlaagd is.

### 3.4 Resultaat

Het resultaat is een Proof-of-Concept-model van de tweede generatie GEN-node waarbij het belang ligt bij verbeterde firmware.

### 3.5 Randvoorwaarden en afbakening

- De firmware is gebaseerd op de laatste versie van de software, geschreven in C.
- De broncode volgt het interne handboek C style guide en de software is opgebouwd in aparte modules.
- Het project richt zich op de software die op de GEN-node draait. De afstudeerder werkt met de aangeboden hardware, namelijk de laatste versie GEN-node.
- In dit project wordt er binnen het Sense2Grow-netwerk voornamelijk gefocust op de GEN-node.

- Elektrotechniek valt buiten de scope van de studie. Voor elektrotechnische vraagstukken, zoals het doen van energiemetingen, kan de hulp van de begeleider of een medewerker gevraagd worden.
- Het project heeft een duur van 17 weken.



## 4. Risicoanalyse

Bij dit project zijn er enkele risico's. Deze worden nader gespecificeerd in tabel 4.1 tot en met tabel 4.5 aan de hand van kansgrootte, impactgrootte en maatregelen.

1. Opdrachtgever heeft gaandeweg het project een ander doel voor ogen
2. Het doel blijkt gaandeweg technisch niet haalbaar te zijn
3. De werkzaamheden blijken complexer dan gepland
4. Hardware raakt defect
5. Er is onvoldoende geheugen op de hardware

Risico	Opdrachtgever heeft gaandeweg het project een ander doel voor ogen
Kans	★☆☆☆☆
Kansverlagende maatregel	Doelen goed vastleggen aan het begin van het project
Impact	★★★★★
Impactverlagende maatregel	Het nieuwe doel proberen in te passen in de requirements
Plan B	Ontwerpen aanpassen en vervolgfases opnieuw doorlopen

» Tabel 4.1

Risico	Het doel blijkt gaandeweg technisch niet haalbaar te zijn
Kans	★☆☆☆☆
Kansverlagende maatregel	Vroegtijdig technisch onderzoek doen
Impact	★★★★★
Impactverlagende maatregel	Overleg om requirements te veranderen waar dit voor belemmering zorgt
Plan B	Het doel aanpassen

» Tabel 4.2

Risico	De werkzaamheden blijken complexer dan gepland
--------	--

Kans	★★★★☆
Kansverlagende maatregel	Vroegtijdig technisch onderzoek doen
Impact	★★★★☆
Impactverlagende maatregel	Enkele requirements naar beneden schalen in prioriteit
Plan B	De complexe werkzaamheden veranderen in minder complexe werkzaamheden en de complexe werkzaamheden opnemen in het advies voor een vervolgproject

» Tabel 4.3

Risico	De GEN-nodes raken defect waardoor er niet getest kan worden
Kans	★★★★☆
Kansverlagende maatregel	Back-up hardware hebben
Impact	★★★★★
Impactverlagende maatregel	Oudere hardware gebruiken
Plan B	Zonder hardware verder werken aan ontwerpen totdat deze weer gereed is

» Tabel 4.4

Risico	Er is onvoldoende geheugen op de hardware
Kans	★★★★☆
Kansverlagende maatregel	Rekening houden bij het ontwerp met de beperkte geheugenruimte
Impact	★★★★☆
Impactverlagende maatregel	Minder gebruikte software delen schrappen uit de firmware
Plan B	Adviseren andere hardware te gebruiken of de firmware op te delen voor verschillende functionaliteiten

» Tabel 4.5

## 5. Aanpak

Dit project wordt verdeeld in zes fases gedurende 17 weken (85 dagen) en wordt uitgevoerd met softwareontwikkelingsmethode Rapid Application Development (RAD).

### 5.1 Methode

Voor het uitvoeren van een project is het belangrijk om een projectmethode te kiezen die past bij het huidige project. Voor het kiezen van een methode is het goed om eerst te bepalen welke ontwikkelstrategie toepasselijk is.

Er zijn vier ontwikkelstrategieën: acceptatie, lineair, iteratief, incrementeel. Dit project bestaat uit drie onderdelen die apart van elkaar ontworpen, ontwikkeld en getest kunnen worden, past dit project binnen een iteratieve en een incrementele strategie. Omdat een incrementele ontwikkelstrategie geschikt is bij een situatie waarbij de ontwikkelaar onbekend is met de gebruikte technologie en een iteratieve strategie de doelstellingen vaag zijn (wat niet het geval is in dit project) is er gekozen voor een incrementele ontwikkelstrategie.

Bij deze ontwikkelstrategie passen enkele methodes: RAD, RUP, Open UP, Scrum, DSDM en XP. Wegens de beperkte grootte van het project (1 ontwikkelaar, 1 opdrachtgever) en de beperkte doorlooptijd (17 weken) is RAD erg geschikt aangezien de focus ligt bij het snel ontwikkelen.

### 5.2 Werkzaamheden

Dit project bevat drie iteraties welke elk een deelvraag moeten gaan beantwoorden.

- Iteratie 1: Hoe kunnen analoge sensoren worden ondersteund met sensorkeuze?
- Iteratie 2: Hoe kunnen sensoren ten behoeve van het meten van luchtkwaliteit worden ondersteund?
- Iteratie 3: Hoe kan er rekening worden gehouden met energiezuinigheid?

Onderstaand tabel (5.1) bevat een globale planning en de bijbehorende producten waarbij het ontwerprapport, de firmware en het testrapport bij elke iteratie aangevuld worden.

Fase	Activiteiten	Resultaat	Duur
<b>Oriëntatiefase</b>	Verhelderen opdracht Risico's analyseren, Planning	Plan van Aanpak	5 dagen
<b>Analysefase</b>	Literatuur bestuderen, Analyse huidige situatie, Requirements achterhalen, Systeemeisen vaststellen	Analyserapport	20 dagen
<i>3 keer itereren</i>	Ontwerpfase	Ontwerpen architectuur	15-20 dagen per iteratie (totaal 50 dagen)
	Ontwikkelfase	Ontwikkel firmware	
	Testfase	Uitvoeren veldtesten	
<b>Adviesfase</b>	Conclusies en adviezen onderbouwen	Adviesrapport	5 dagen

» Tabel 5.1, werkzaamheden

Parallel wordt er gewerkt aan het afstudeerdossier. Hier wordt 15 dagen voor uitgetrokken, die verdeeld worden over de gehele periode. Deze wordt uiterlijk de 17<sup>e</sup> week ingeleverd op 5 juni 2020.

## 5.3 Planning

In onderstaand tabel (5.2) is een globale planning opgenomen.

Week	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
<b>Oriëntatiefase</b>																	
<b>Analysefase</b>																	
<b>Iteratie 1</b>																	
<b>Iteratie 2</b>																	
<b>Iteratie 3</b>																	
<b>Adviesfase</b>																	

» Tabel 5.2, Planning

Bijlage B

# Analyserapport

# Verbeterde firmware GEN-node



## Analyserapport

*Maart 2020, Gouda*

**Opdrachtgever**

Olaf Peters

**Auteur**

Jeremy Hut



DE HAAGSE  
HOGESCHOOL

# Inhoudsopgave

<b>1. Inleiding</b>	<b>2</b>
<b>2. Huidige GEN-node</b>	<b>3</b>
2.1 Sense2Grow-netwerk	3
2.1.1 LoRa	3
2.2 PCB	4
2.2.1 LoRa-module	5
2.2.2 Analog Digital Converter	6
2.2.3 I <sup>2</sup> C	8
2.2.4 SPI	8
2.2.5 DIP-switches	8
2.2.6 Voeding	9
2.2.7 Pin layout	9
2.3 Software	10
<b>3. Onderzoek gewenste firmware GEN-node</b>	<b>13</b>
3.1 Analoge sensoren universeel ondersteunen	13
3.2 Geselecteerde luchtkwaliteitssensoren ondersteunen	16
3.3 Energieverbruik verbeteren	17
<b>4. Requirements</b>	<b>20</b>
4.1 Functionele requirements	20
4.2 Niet-functionele requirements	20
<b>5. Systeemeisen</b>	<b>21</b>
<b>Literatuur</b>	<b>22</b>
<b>Bijlage 1 - Technische tekening GEN-node</b>	<b>23</b>

# 1. Inleiding

In het analyserapport wordt het onderzoek beschreven wat nodig is voor GEN-node versie 3. Deze node wordt gebruikt voor het Sense2Grow-netwerk om sensorgegevens te versturen. Het analyserapport bestaat ten eerste uit een onderzoek naar wat de huidige situatie is van de GEN-node. Ook wordt er beschreven wat belangrijke componenten zijn en onderzoek naar elementen die mogelijk gebruikt worden in de nieuwe GEN-node. Daarna zijn de requirements te vinden. Deze zijn ingedeeld op prioriteit, waarbij 'must have' de hoogste prioriteit is. Tot slot worden er systeemeisen gedefinieerd.

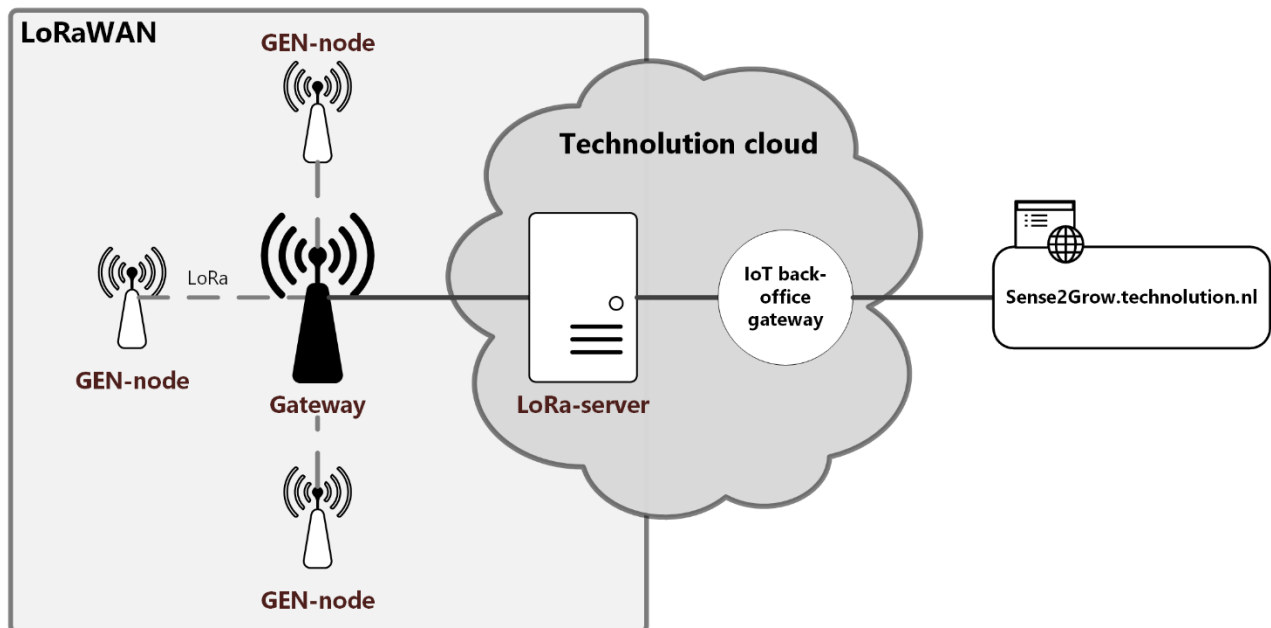


## 2. Huidige GEN-node

De huidige GEN-node is de tweede generatie. Deze versie heeft ten opzichte van de eerste generatie aanpassingen gehad op het gebied van hardware, waarbij er voorbereidingen zijn gedaan voor de ondersteuning van analoge sensoren.

### 2.1 Sense2Grow-netwerk

Sense2Grow is het overkoepelende project waarin de GEN-node werkt. Sense2Grow is een netwerk dat onder andere gemaakt is voor het gebruik in kassen. Het netwerk heeft momenteel als doel een optimalisatie van plantengroei en energieverbruik. Bij de klant zijn er sensoren geplaatst. Elke sensor wordt uitgelezen door een node, de GEN-node. Deze stuurt de sensordata op via het draadloze communicatieprotocol LoRa (zie ook 2.1.1 LoRa). Bij de klant zijn ook LoRa-gateways geïnstalleerd die de sensordata over het internet verzendt naar de servers van Technolution. In afbeelding 2.1 is dit uitgebeeld. De data is online inzichtelijk gemaakt in een realtime dashboard gemaakt op [sense2grow.technolution.nl](https://sense2grow.technolution.nl), zoals helemaal rechts te zien is in de afbeelding.



» Afbeelding 2.1, netwerk Sense2Grow

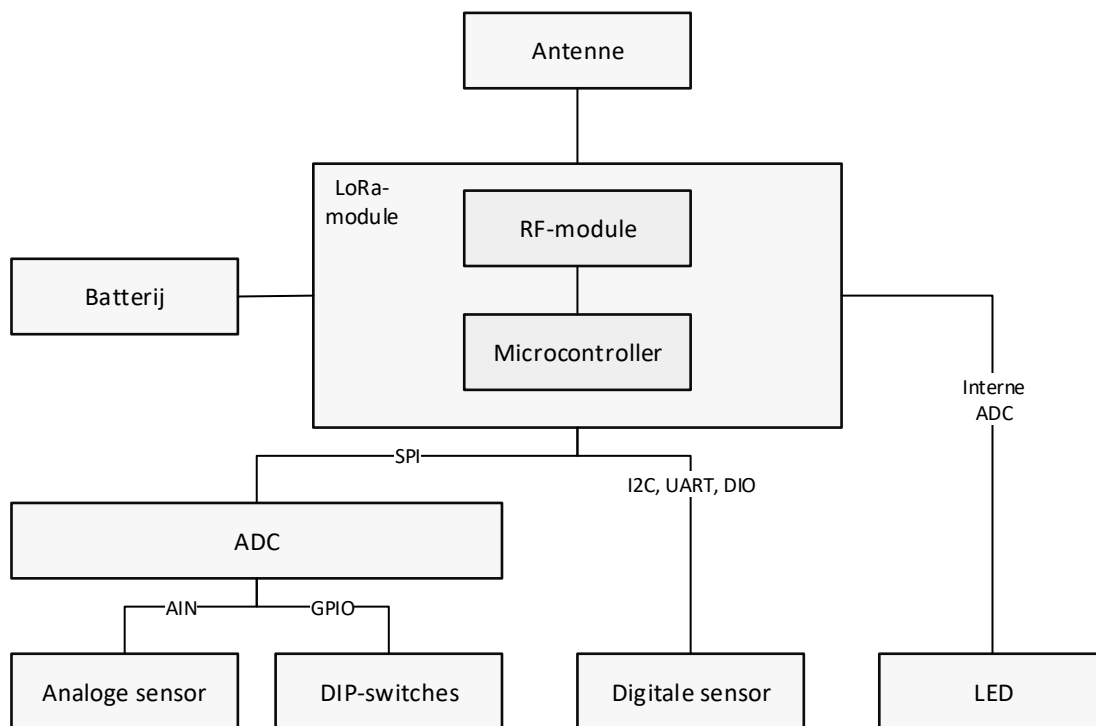
#### 2.1.1 LoRa

LoRa is een draadloos communicatieprotocol. De huidige GEN-node maakt hier gebruik van. LoRa staat voor Long Range en is geschikt om over lange afstanden gegevens te versturen door middel van radiografische signalen. [1] Deze gegevens kunnen verstuurd worden met laag energieverbruik. Dit maakt het geschikt voor apparaten die werken op batterij, zoals de

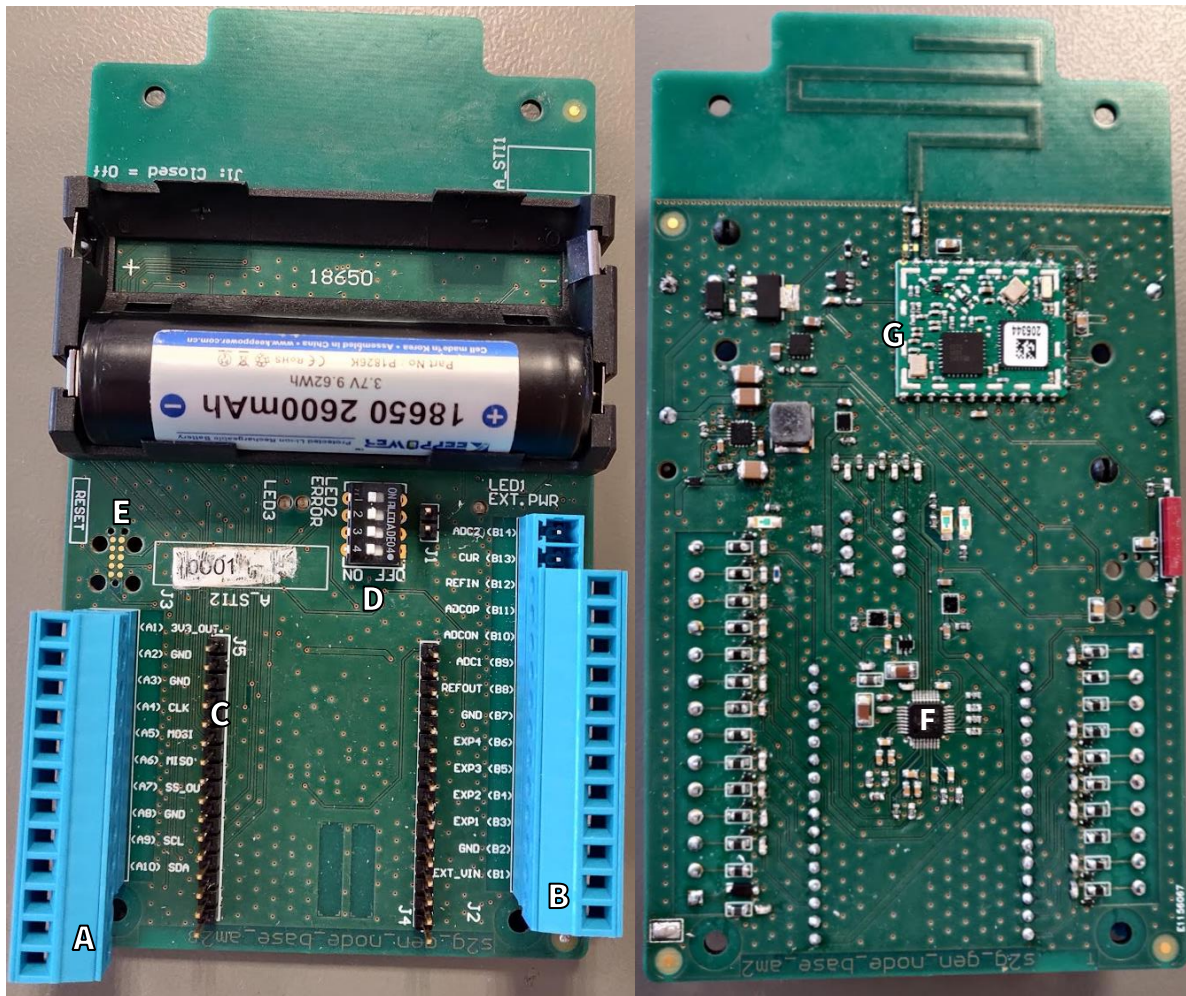
GEN-node. Het protocol dat op de communicatielaag werkt is LoRaWAN. Een node maakt verbinding met een gateway. De gateway wordt ontsloten door middel van een internetverbinding.

## 2.2 PCB

De GEN-node is een Printed Circuit Board (PCB, ook wel bekend als printplaat) waarop modules en sensoren aangesloten kunnen worden. De PCB is een ontwerp van Technolution. De microcontroller van deze PCB is te vinden in de LoRa-module, hier staat de complete firmware op.



» Afbeelding 2.1, hardware-overzicht PCB



» Afbeelding 2.2, boven- en onderaanzicht PCB

#### Legenda afbeelding 2.2

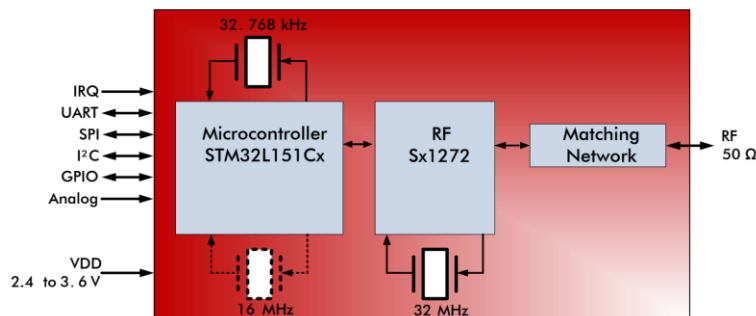
- A. Schroefconnectoren A-zijde
- B. Schroefconnectoren B-zijde
- C. Expansieheader
- D. DIP-switches
- E. Programmeerheader
- F. ADS114S06B ADC
- G. WiMOD iM880B LoRa-module

#### 2.2.1 LoRa-module

De LoRa-module die op de GEN-node gebruikt wordt is de WiMOD iM880B. Het bevat een microcontroller: STM32L151Cx. Deze microcontroller bevat ook gelijk de aangepaste firmware van de GEN-node. Hiernaast werkt een RF transceiver: Sx1272 De antenne van de module zit verwerkt in de PCB. [2]

Flash	128 kB
RAM	32 kB
Stroom in idle-stand	5 mA
Stroom tijdens ontvangen	11.2 mA
Stroom tijdens verzenden	117-126 mA

» Tabel 2.1, kenmerken LoRa Module [3]



» Afbeelding 2.3, schematische tekening iM880B LoRa-module [3]

### 2.2.2 Analog Digital Converter

Voor het omzetten van analoge signalen naar digitale signalen wordt een Analog Digital Converter (ADC) gebruikt. De spanning op de ADC wordt omgezet naar bits. De waarde hiervan is de grootte van deingangsspanning ten opzichte van de referentiespanning [4]. Het verschil tussen de minimale en maximale waardes heet de resolutie.

De waarde hiervan laat zich berekenen aan de hand van de volgende vergelijking: [5]

$$\frac{\text{Resolutie ADC}}{\text{Referentiespanning}} = \frac{\text{ADC-waarde}}{\text{Gemeten spanning}}$$

Een voorbeeld voor deze vergelijking om de ADC-waarde om te zetten naar de spanning van de sensor. Stel we hebben een 10-bit ADC. Neem aan dat de resolutie van de ADC in dit voorbeeld 1023 is ( $= 2^{10} - 1$ ) en de referentiespanning 5V is. De ADC-waarde is op een bepaald moment 434. De vergelijking ziet er dan zo uit:

$$\frac{1023}{5 \text{ V}} = \frac{434}{\text{Gemeten spanning}}$$

$$\text{Gemeten spanning} = 5 \text{ V} * \frac{434}{1023} = 2.12 \text{ V}$$

Het is ook mogelijk om de stroom te meten. Op de PCB is hiervoor een input beschikbaar met een weerstand van 100 ohm. Sluit deze input aan met de normale ADC2-input die ook

aangesloten is aan de sensor. Gebruik vervolgens  $I = V/R$ , waarbij R de bekende 100 ohm is en V de gemeten spanning.

Er is een interne ADC beschikbaar op de LoRa-module, maar deze biedt niet in alle gevallen voldoende precisie. Voor de GEN-node was er daarom in een eerder project gekozen voor een ADS114S06B ADC van Texas Instruments. Dit is een 16-bits ADC met 6 kanalen [4]. Deze externe ADC wordt aangestuurd door middel van SPI. De commando's zijn te vinden in tabel 2.2 en de registers in tabel 2.3.

COMMAND	DESCRIPTION	FIRST COMMAND BYTE	SECOND COMMAND BYTE
<b>Control Commands</b>			
NOP	No operation	0000 0000 (00h)	—
WAKEUP	Wake-up from power-down mode	0000 001x (02h, 03h) <sup>(1)</sup>	—
POWERDOWN	Enter power-down mode	0000 010x (04h, 05h) <sup>(1)</sup>	—
RESET	Reset the device	0000 011x (06h, 07h) <sup>(1)</sup>	—
START	Start conversions	0000 100x (08h, 09h) <sup>(1)</sup>	—
STOP	Stop conversions	0000 101x (0Ah, 0Bh) <sup>(1)</sup>	—
<b>Calibration Commands</b>			
SYOCAL	System offset calibration	0001 0110 (16h)	—
SYGCAL	System gain calibration	0001 0111 (17h)	—
SFOCAL	Self offset calibration	0001 1001 (19h)	—
<b>Data Read Command</b>			
RDATA	Read data	0001 001x (12h / 13h) <sup>(1)</sup>	—
<b>Register Read and Write Commands</b>			
RREG	Read <i>nnnn</i> registers starting at address <i>rrrr</i>	001r <i>rrrr</i> <sup>(2)</sup>	000n <i>nnnn</i> <sup>(3)</sup>
WREG	Write <i>nnnn</i> registers starting at address <i>rrrr</i>	010r <i>rrrr</i> <sup>(2)</sup>	000n <i>nnnn</i> <sup>(3)</sup>

(1) x = don't care.

(2) r *rrrr* = starting register address.

(3) n *nnnn* = number of registers to read or write – 1.

» Tabel 2.2, Commando's ADC [6]

ADDR	REGISTER	DEFAULT	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
00h	ID	xxh	RESERVED					DEV_ID[2:0]		
01h	STATUS	80h	FL_POR	RDY	0	0	0	0	0	0
02h	INPMUX	01h	MUXP[3:0]				MUXN[3:0]			
03h	PGA	00h	0	0	0	PGA_EN[1:0]		GAIN[2:0]		
04h	DATARATE	14h	0	CLK	MODE	1	DR[3:0]			
05h	REF	10h	0	0	REFP_BUF	REFN_BUF	REFSEL[1:0]		REFCON[1:0]	
06h	IDACMAG	00h	0	0	0	0	IMAG[3:0]			
07h	IDACMUX	FFh	I2MUX[3:0]				I1MUX[3:0]			
08h	VBIAS	00h	0	VB_AINC	VB_AIN5	VB_AIN4	VB_AIN3	VB_AIN2	VB_AIN1	VB_AIN0
09h	SYS	10h	SYS_MON[2:0]			CAL_SAMP[1:0]		TIMEOUT	0	0
0Ah	RESERVED	00h	RESERVED							
0Bh	OFCAL0	00h	OFC[7:0]							
0Ch	OFCAL1	00h	OFC[15:8]							
0Dh	RESERVED	00h	RESERVED							
0Eh	FSCAL0	00h	FSC[7:0]							
0Fh	FSCAL1	40h	FSC[15:8]							
10h	GPIODAT	00h	DIR[3:0]				DAT[3:0]			
11h	GPIOCON	00h	0	0	0	0	CON[3:0]			

» Tabel 2.3, Registers ADC [6]

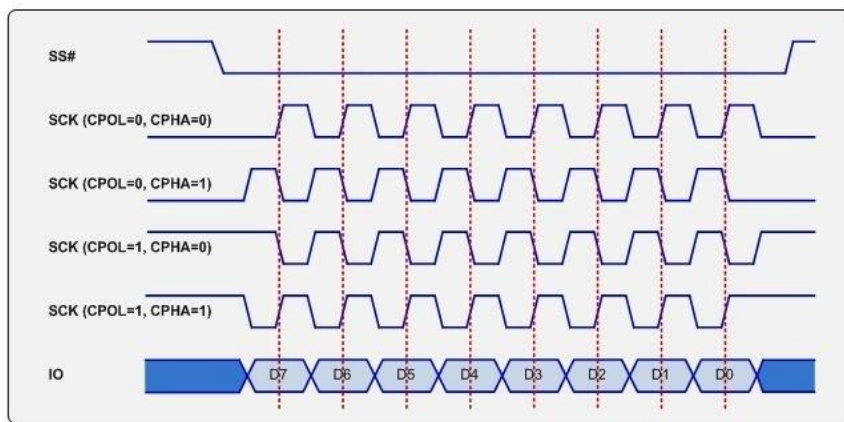
### 2.2.3 I<sup>2</sup>C

I<sup>2</sup>C, of Two Wire Interface (TWI), is een communicatieinterface om een sensor mee te verbinden. Dit wordt ondersteund door de node. Voor I<sup>2</sup>C is er een specifieke configuratie voor de meeste sensoren nodig. De sensoren die momenteel toegepast worden met I<sup>2</sup>C heten co2\_lp8, particulatematter\_sps30, celcius\_sh31, dew\_hs3s en celcius\_ds18b20.

### 2.2.4 SPI

Serial Peripheral Interface (SPI) is net als I<sup>2</sup>C een interface voor communicatie. Ook hier is per sensor een aparte configuratie nodig. Er zijn momenteel geen sensoren in gebruik die gebruik maken van de SPI. De nieuwe ADC maakt wel gebruik van de SPI.

SPI-communicatie van master naar een slave heeft 4 lijnen nodig. Door de lijn Slave Select ( $\overline{SS}$ ) naar een logische nul te zetten wordt er een slave geselecteerd waarmee de master kan gaan communiceren. De serial clock (SCK) vertelt de slave wanneer er een sample van de data moet worden genomen. Voor deze ADC is dit het moment dat de clock van een logische 1 naar een 0 gaat (mode 1). De benodigde mode kan per slave verschillen. Afbeelding 2.4 laat dit zien. Hierbij vertegenwoordigen de SCK-lijnen de serial clock lijnen met van boven naar beneden mode 1 tot 4. De rode verticale lijn toont de samplemomenten.



» Afbeelding 2.4, communicatiesignalen SPI (bron: <https://www.corelis.com/education/tutorials/spi-tutorial/>)

### 2.2.5 DIP-switches

De PCB heeft een viertal DIP-switches. Dit zijn kleine, handmatige schakelaars. Ze hebben 2 standen. Zo zijn er 16 mogelijkheden. Deze zijn op de PCB gezet met het doel een sensortype te kunnen selecteren. Momenteel worden de switches nog niet gebruikt in de firmware. De switches zijn aangesloten op de 4 GPIO-ingangen van de ADC. Voordat de aangesloten sensor wordt uitgelezen, moet zou de ADC dus eerst de waarde van de DIP-switches moeten hebben uitgelezen. De pin layout is te vinden in hoofdstuk 2.2.7.



## 2.2.6 Voeding

De node heeft ruimte voor 2 Li-ion-batterijen (3.7V) [2] van het type 18650. Het is ook een optie om een externe voeding van 5V tot 24V aan te sluiten [4]. De microcontroller heeft 3,3 V nodig voor voeding.

## 2.2.7 Pin layout

De pin layout van de GEN-node is te vinden in onderstaand tabel.

Pin			Functie		
<b>A1</b>	VIN	Voeding	<b>B1</b>	3v3_Switched	3V3 schakelbare voeding
<b>A2</b>	GND	Voeding	<b>B2</b>	GND	Ground
<b>A3</b>	EXP1	Custom Add-on	<b>B3</b>	GND	Ground
<b>A4</b>	EXP2	Custom Add-on	<b>B4</b>	P9/SPI_CLK	SPI of DIO
<b>A5</b>	EXP3	Custom Add-on	<b>B5</b>	P8/SPI_MOSI	SPI of DIO
<b>A6</b>	EXP4	Custom Add-on	<b>B6</b>	P7/SPI_MISO	SPI of DIO
<b>A7</b>	GND	Ground	<b>B7</b>	P14 / SPI_SS_OUT / DIO	SPI of DIO
<b>A8</b>	REFOUT / EC5Supply	2.5V / 10mA schakelbare uitgang	<b>B8</b>	GND	Ground
<b>A9</b>	ADC1 / EC5Data	0-2,5V ingang	<b>B9</b>	P12/I2C_SCL	I2C of DIO
<b>A10</b>	ADC0N / RTD_L1	Diff. ingang - / RTD aansluiting	<b>B10</b>	P13/I2C_SDA	I2C of DIO
<b>A11</b>	ADC0P / RTD_L2	Diff. ingang + / RTD aansluiting			
<b>A12</b>	REFIN / RTD_L3	RTD aansluiting			
<b>A13</b>	RCUR	Enable 4-20mA pin			
<b>A14</b>	ADC2	0-2,5V ingang / 4-20mA ingang			

» Tabel 2.4, pin layout PCB

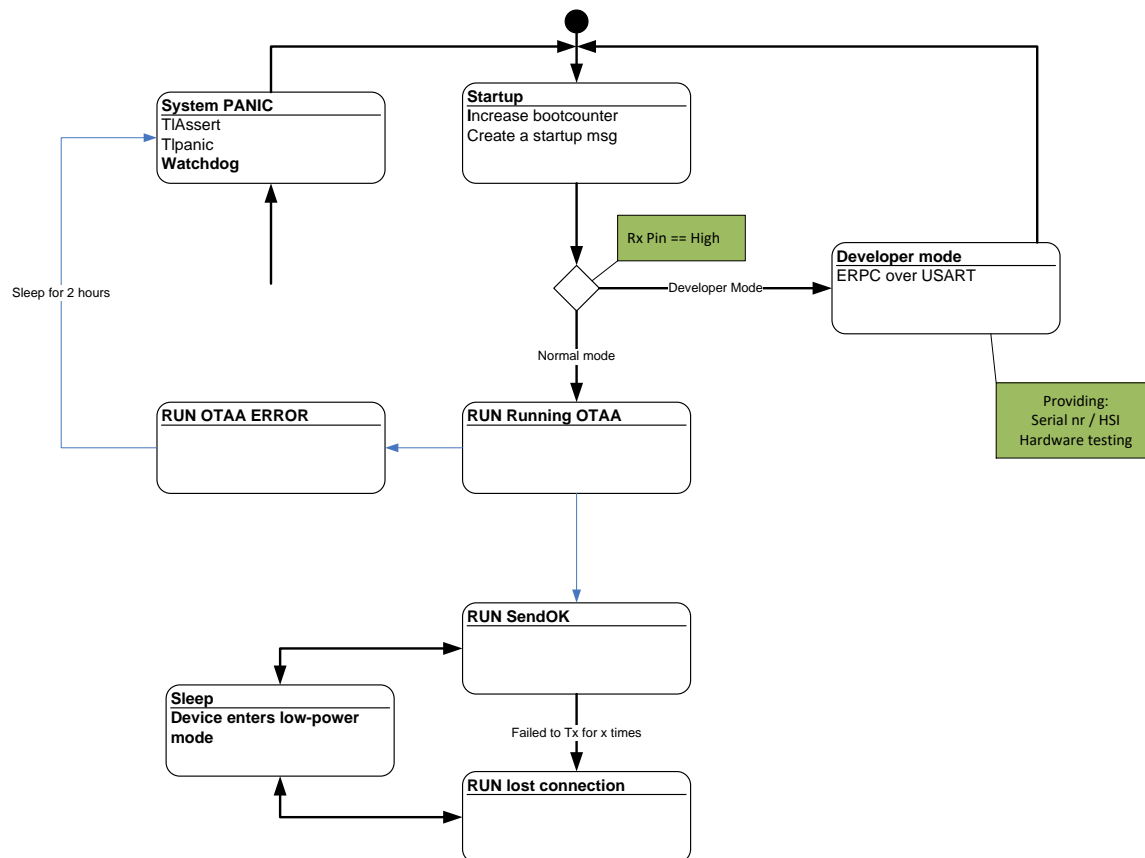
Voor de vier GPIO-ingangen op de ADC geldt de volgende functies:

ADC_	Functie
<b>GPIO0</b>	DIP-switch pin 1
<b>GPIO1</b>	DIP-switch pin 2
<b>GPIO2</b>	DIP-switch pin 3
<b>GPIO3</b>	DIP-switch pin 4

» Tabel 2.5, GPIO-verbindingen op de ADC

## 2.3 Software

De huidige software is opgebouwd om 1 sensortype uit te lezen. Bij het uploaden van de firmware naar de GEN-node wordt er gekozen voor welke sensor dit moet zijn. Dit specificeert hoe de node moet communiceren met de sensor en hoe deze waarde wordt omgerekend naar bijvoorbeeld een temperatuur.



» Afbeelding 2.5, State machine diagram [7]

### State machine diagram (afbeelding 2.5)

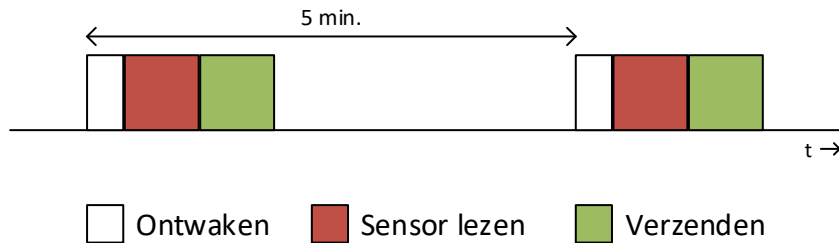
De node start op (Startup), vervolgens probeert het verbinding te maken met een gateway (RUN Running OTAA). Als dit is gelukt stuurt het zijn eerste meetwaarde(n) (RUN SendOk). Als dit lukt valt het apparaat in slaap (Sleep). Na vijf minuten zullen er opnieuw gegevens verzonden worden (RUN SendOk). Mocht er 2 uur lang geen gateway gevonden kunnen worden of blijven er bepaalde dingen mislukken, komt het systeem in de toestand System PANIC, waarna het zich herstart.

### Duty cycle

Aangezien de node niet hoeft te luisteren of er nog pakketten beschikbaar zijn om ontvangen te worden is de duty cycle best eenvoudig (afbeelding 2.6). Iedere vijf minuten ontwaakt de



node en wordt er data verzonden. Het ontwaken tot en met het verzenden duurt ongeveer 100ms. Mocht het pakketje niet aankomen, dan zal het pas 5 minuten later weer opnieuw geprobeerd worden. Dit heeft te maken met een ontwerpkeuze die gemaakt was voor het besparen van energie.



» Afbeelding 2.6, Duty Cycle voor het uitlezen sensor

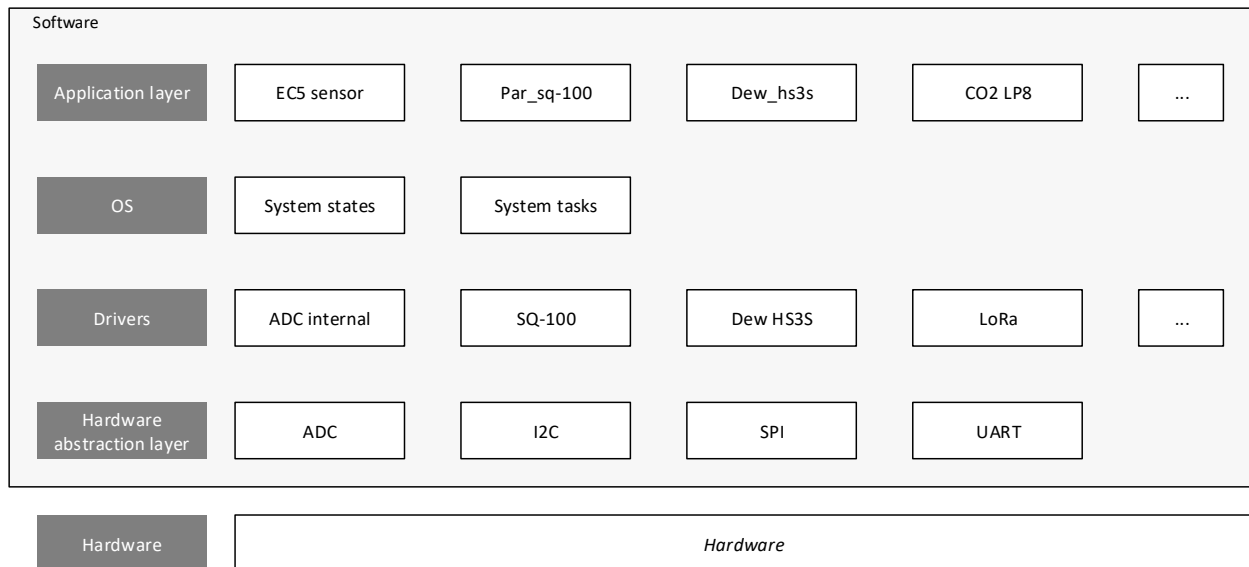
## Taken

In het systeem zijn er verschillende taken die aangeroepen worden.

- SysTick Task
  - Ontwaakt het apparaat en start andere taken
- Sensor Task
  - Regelt dat de sensorwaardes uitgelezen worden, maakt een bericht en plaatst in buffer
- System Monitor Task (SysMon)
  - Monitort het apparaat en stuurt apparaatstatistieken zoals batterijwaarde en interne temperatuur
- Transceiver Task
  - Verzend berichten in buffer door LoRa

## Architectuur

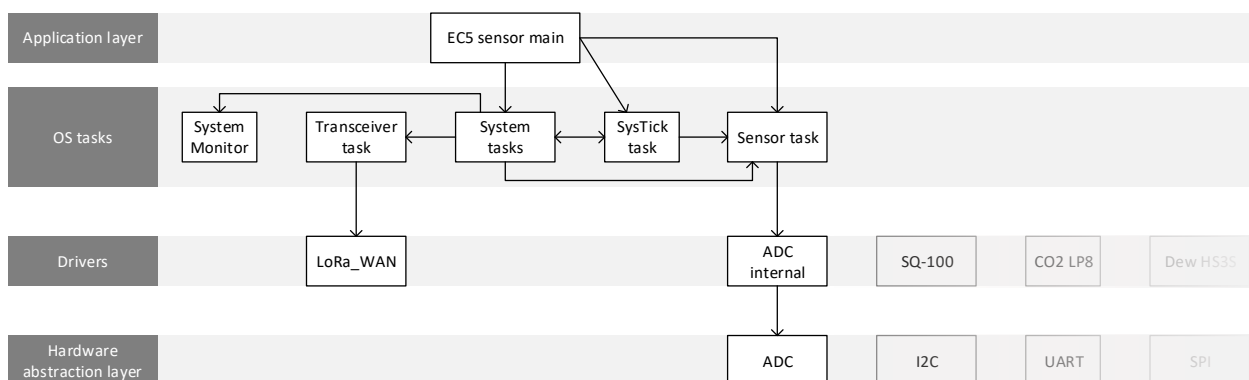
De software bestaat momenteel uit de lagen en elementen getoond in afbeelding 2.7. De elementen stellen geen klassen voor aangezien dit niet bestaat in de gebruikte programmeertaal C. De afbeelding laat een vereenvoudigd totaaloverzicht zien van de software.



» Afbeelding 2.7, overzicht software

Elk element in de applicatielaag bevat een main-functie. Hiermee kan de software aangepast worden voor een bepaalde sensor. Bij het uploaden naar de node wordt er gekozen welke applicatie bij het apparaat hoort. Als er een andere sensor aangesloten wordt, moet er dus nieuwe firmware geüpload worden.

De onderlinge relaties laten zich zien in afbeelding 2.8 als de firmware voor een sensor wordt toegepast. Als voorbeeld is er gekozen voor de EC5-sensorapplicatie. Hierdoor is er een specifieke main-functie en een specifieke sensortask afgestemd op deze sensor.



» Afbeelding 2.8, elementen en relaties EC5-software

## 3. Onderzoek gewenste firmware

### GEN-node

#### 3.1 Analoge sensoren universeel ondersteunen

Voor de nieuwe firmware is het gewenst om analoge sensoren te ondersteunen zonder dat er voor iedere sensor een specifiek stuk software is geschreven. Een generieke ondersteuning voor een groot aantal analoge sensoren is gewenst. Dit zal moeten gebeuren met behulp van de nieuwe ADC. Een analoge sensor toont zijn resultaat door de stroom of de spanning van het uitgangssignaal te veranderen. De schaal waarbinnen de minimale en maximale waardes vallen kunnen verschillen per type sensor. Binnen de industrie zijn er ook standaarden voor deze waardes, namelijk sensoren met een uitgangssignaal van 4 tot 20 mA, 0 tot 5 V en 0 tot 10 V.

Omdat we te maken hebben met verschillende meetmethodes per type analoge sensor, moet er een keuze kunnen worden gemaakt hoe er uitgelezen moet worden. Hiervoor moet er voor diverse analoge sensortypes een voorinstelling worden ingesteld waarbij er dus gekozen kan worden op welke manier de sensor uitgelezen moet worden. In een eerder project is er gekozen om dit te doen door middel van DIP-switches. Deze kunnen in een stand worden gezet die een bepaalde voorinstelling vertegenwoordigen.

In datzelfde project is er een nieuwe ADC toegevoegd. Dit is een externe ADC. Hiervoor is nog geen goede software geschreven. Er is alleen software gemaakt om te testen of de ADC werkt. De ADC wordt aangestuurd via een bestaande SPI-driver.

#### **Relevante parameters binnen de ADC**

Een ADC meet alleen spanning. In het geval dat er een stroom gemeten moet worden moet pin RCUR aangesloten worden aan de ingangspin (ADC2). Bij RCUR is een weerstand geplaatst die ervoor zorgt dat de gemeten spanning omgerekend kan worden naar stroom (met de wet van Ohm). De positieve referentiespanning kan ook bepaald worden (pin REFIN aansluiten). Dit heeft invloed op de minimale en maximale waardes. In plaats van een referentiespanning op deze pin aan te sluiten kan er ook gebruik gemaakt worden van de interne spanning van 2,5V als referentiespanning. Ook kan de resolutie (in Volt) verkleind worden door middel van de programmable gain amplifier (PGA). Hierdoor kunnen kleinere spanningen 'uitvergroot' worden. De kleinste resolutie die mogelijk is, is 0,020V met een gain van 128 en de interne referentiespanning van 2,5V.

De analoge ingangen van de ADC kunnen een spanning aan van tussen de -0,6V en 5,8V [6]. Zie ook afbeelding 3.1.

### Analoge sensoren en voorinstellingen

Er is een lijst opgesteld met sensoren die voor kunnen komen, zie tabel 3.2. Deze lijst bestaat uit sensoren die gebruikt zijn bij

klanten, waarvoor interesse is bij klanten of waarvoor onderzoek is gedaan in eerdere projecten. Deze lijst bevat ook analoge sensoren. Voor de sensoren met een maximale uitgangsspanning van 3,3V, 5V en 10V kan er voor de referentiespanning gekozen worden voor het aanleveren van een spanning gelijk aan de maximale uitgangsspanning. 3,3V en 5V zou wegens het geringe verschil ook beiden aangesloten kunnen worden op een referentiespanning van 5V. Voor de maximale uitgangsspanningen van 2.5V en lager kan er gekozen worden voor de interne referentiespanning.

GAIN SETTING	FSR
1	$\pm 2.5 \text{ V}$
2	$\pm 1.25 \text{ V}$
4	$\pm 0.625 \text{ V}$
8	$\pm 0.313 \text{ V}$
16	$\pm 0.156 \text{ V}$
32	$\pm 0.078 \text{ V}$
64	$\pm 0.039 \text{ V}$
128	$\pm 0.020 \text{ V}$

» Tabel 3.1, Gain-waardes ( $\text{FSR} = \pm V_{\text{ref}} / \text{gain}$ )

Sensor	Functie	Communicatie	Uitgang
PT500	Temperatuur	Analoog	$401.5\Omega - 1568\Omega$ (460-620 $\Omega$ )
EC5	Bodemvocht	Analoog	$V_{\text{cc}}(10\% - 50\%) \text{ V}$
SQ-500/SQ-100	PAR (licht)	Analoog	0 – 40 mV
Licor LI-190	PAR	Analoog	0-0.050mA – 0-0.100mA
TZW-330 18-03 (AliE.)	Luchtdrukverschil	Analoog	4 – 20 mA
SMP6-a	Pyranometer	Analoog	4 – 20 mA
-	Dummy	Analoog	0 – 10 V
Ds18b20	Temperatuur	1-wire	
Sh31	Temperatuur	I2C	
LP8	CO2	I2C	
Hs3s	Dauw	I2C	
SPS30	PM	I2C	
RE46C190	Rook	GPIO	
NO2-B43F	NO2	Analoog	0 - 5 V (2x)
ME2-O2- $\Phi$ 20	O2	Analoog	0 - 3.3 V
SO2-B4	SO2	Analoog	0 - 5 V (2x)

» Tabel 3.2, Lijst van sensoren

Voor een spanning van 10V is deze ADC ongeschikt. Op de analoge ingangen is een maximale spanning van 5,8V mogelijk. Dit betekent dat de uitgangsspanning van de sensor naar beneden gebracht moet worden. Hiervoor zou gekozen kunnen worden voor 5V of 2,5V. Het voordeel van 2,5V is dat dit een veilige marge geeft wanneer de gebruiker per ongeluk een grotere spanning aanbiedt, aangezien deze spanningen niet groter mogen zijn dan 5,8V.

Voor sensoren die variëren in stroom geldt een andere aanpak. Hiervoor moet een bekende weerstand worden aangesloten. Door de stroom te vermenigvuldigen met de weerstandswaarde wordt het inzichtelijk wat de ingangsspanning zal worden. Voor de 4 – 20 mA sensoren zou dit met de ingebouwde weerstand op de PCB van 100 ohm betekenen dat de maximale uitgangsspanning 2V is. Hiervoor is geen gain nodig. Een weerstand van 120 ohm zou voor een uitgangsspanning van 2,4V voor een grotere precisie zorgen, maar dan zou de ingebouwde weerstand vervangen moeten worden of 20 ohm worden toegevoegd. Voor zwakkere signalen is een grotere gain nodig en/of een grotere weerstandswaarde.

Het lijkt dus nodig dat er met de DIP-switches gewisseld moet kunnen worden tussen de interne referentiespanning van 2,5V en een referentiespanning aangesloten op de pin REFIN. Als de referentiespanning gebruikt wordt is het voor sommige sensoren nodig om een gain in te stellen. Aan de hand van de te gebruiken sensoren, de aanbevolen weerstanden en de spanningsresolutie is de gain bepaald in tabel 3.3. Een gain van 1 (geen), 2, 8 en 32 komen voor in deze tabel. Deze zijn in ieder geval nodig om op de DIP-switches instelbaar te maken. Hoe beter de gain past bij de spanningsresolutie, hoe preciezer het resultaat. Afhankelijk van de wensen van de opdrachtgever en de vrije plekken op de DIP-switch kunnen er meer gain-instellingen worden gebruikt. De DIP-switches bieden 16 mogelijkheden.

### Hardware-aanpassingen

Voor het naar beneden brengen van 10 volts ingangsspanning was het nodig om een hardware-aanpassing uit te voeren. Om de interface eenvoudig te houden is er gekozen voor extra DIP-switches waarmee in dit geval hardwarematig tussen schakelingen kan worden gewisseld. Ook 5 volts spanning kan hiermee naar 2,5 V gebracht. Deze keuze is gemaakt zodat er altijd gebruik gemaakt kan worden van de interne referentiespanning (2,5 V) en is het niet nodig om via alternatieve spanningsbronnen nog referentiespanningen aan te leveren. Er kan ervoor gekozen worden om de optie voor het kiezen van een externe referentiespanning op de DIP-switches (softwarematig) te laten vervallen.

Bij deze hardware-aanpassing wordt er direct een andere soort antenne gebruikt. Het gaat om een helical antenne en is één van de aanbevelingen uit het voorgaande project. Deze spiraalvormige antenne zou het zendbereik moeten verbeteren.

Sensor	Uitgang	Weerstand	V <sub>resolutie</sub>	Gain
PT500	460-620 $\Omega$	-	0,31 V	8
EC5	2,5(10% - 50%) V	-	1,25 V	2
SQ-500/SQ-100	0 – 40 mV	-	0,040 V	32
Licor LI-190	0-0.050 mA – 0-0.100 mA	604 $\Omega$	0,0604 V	32
TZW-330 18-03 (AliE.)	4 – 20 mA	100 $\Omega$	2 V	1

SMP6-a	4 – 20 mA	100 $\Omega$	2 V	1
-	0 – 10 V	-	2,5 V	1

» Tabel 3.3, Sensoren en gain-instelling. Referentiespanning is 2,5 V (intern)

### Samenvatting

- Gewenst is een universele ondersteuning van analoge sensoren op de nieuwe ADC
- Er zijn voorinstellingen nodig, hiervoor zijn DIP-switches.
- Gain heeft de waarde 1, 8 en 32 nodig en eventueel nog meer waarden om te kiezen op de DIP-switch.
- 10V spanningen voor de analoge ingangen moet naar beneden gebracht worden.
- Wisselen tussen interne (2,5 V) en externe referentiespanning (VREF) met de DIP-switch of altijd kiezen voor de interne referentiespanning.

## 3.2 Geselecteerde luchtkwaliteitssensoren ondersteunen

Er zijn twee sensoren gekocht die onderdeel zijn van het meten van luchtkwaliteit in het Sense2Grow-netwerk. Het gaat om een sensor die stikstofdioxide meet ( $\text{NO}_2$ ) en een sensor die zwaveldioxide meet ( $\text{SO}_2$ ). De types zijn NO2-B43F en SO2-B4. Beide sensoren zijn analoge sensoren. Hierdoor zouden ze aangesloten kunnen worden als universele analoge sensor. Het verschil is dat deze sensoren niet één maar twee uitgangssignalen heeft. Samen met het ingangssignaal van de sensor levert dit een driedraads sensor op. Beide sensoren zijn gemonteerd op een Alphasense B4 Individual Sensor Board (ISB).

Een driedraadssensor levert een nauwkeurigere weergave omdat hiermee temperatuursinvloeden op de draden kan worden weggefilterd. Op het eerste draad wordt de voeding aangesloten. Op het tweede draad wordt de draadsweerstand gemeten. Op het derde draad wordt de meting en de draadsweerstand gemeten. Wordt de metingswaarde van de draadsweerstand afgetrokken, dan is het resultaat een meting zonder de invloeden van de draadsweerstand. [8]

	NO2-B43F	SO2-B4
<b>Functie</b>	NO2	SO4
<b>Communicatie</b>	Analoog	
<b>Uitgangssignaal</b>	0 – 5 V (wordt 0 – 2,5 V met converter)	
<b>Referentiespanning</b>	2,5 V	
<b>Gain</b>	1	

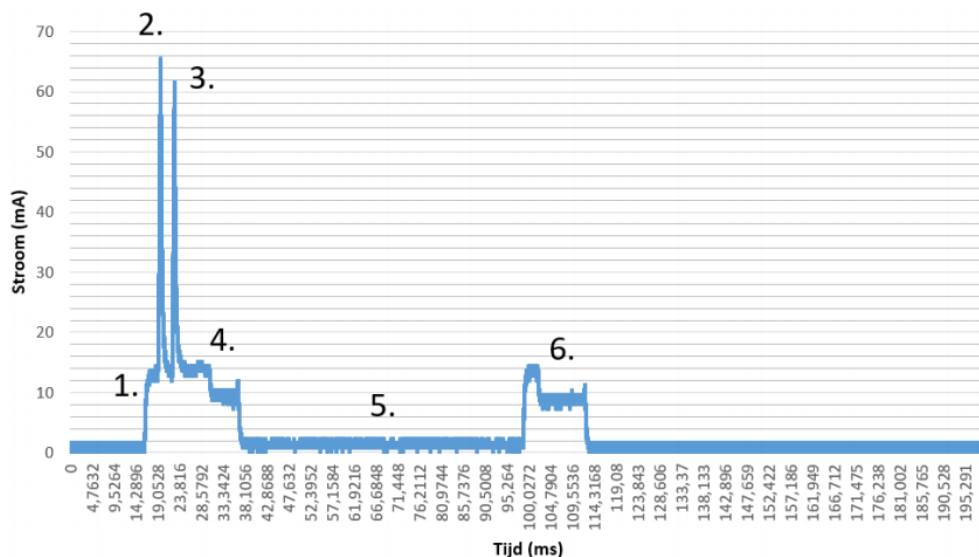
» Tabel 3.4, Eigenschappen NO2-B43F en SO2-B4

### 3.3 Energieverbruik verbeteren

Momenteel is de testsoftware die geschreven is voor de ADC niet optimaal wat betreft het energieverbruik volgens de opdrachtgever. Op hardwaregebied zijn er op verschillende vlakken al ontwerpkeuzes gemaakt die het energieverbruik ten goede komen. Zo worden verschillende onderdelen spanningsloos gemaakt wanneer de node in slaap valt. Dit is de volledige ADC en de referentiespanningen [4].

Het energieverbruik bij het gebruik van een sensor op de externe ADC is te zien in afbeelding 3.1. Hierin zijn de volgende gebeurtenissen te zien: [4]

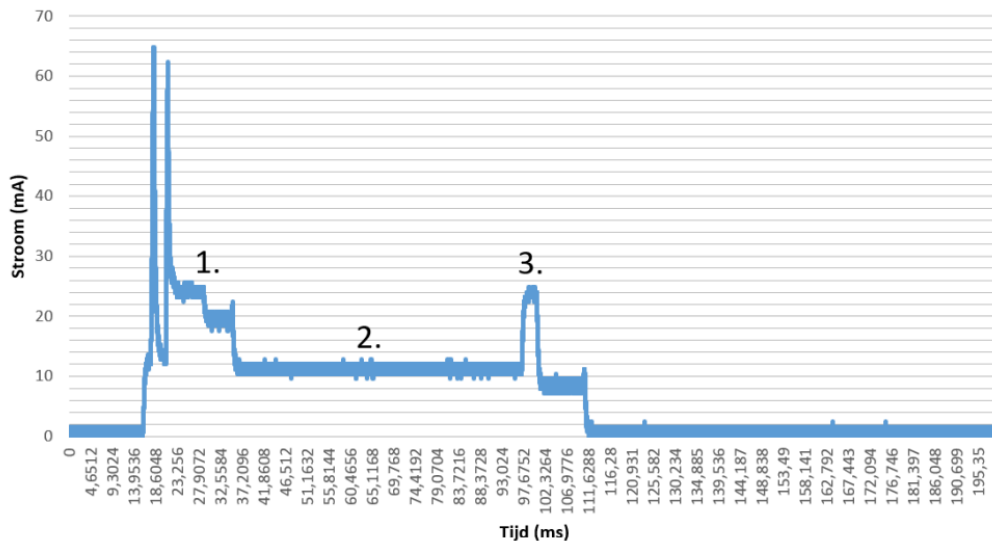
1. Systeem wordt ontwaakt
2. De spanning over de ADC wordt ingeschakeld. Dit veroorzaakt een stroompiek door het laden van de condensatoren
3. De referentiespanning wordt ingeschakeld. Hier is ook weer een piek te zien door de condensator
4. De microcontroller stuurt een START commando naar de ADC en gaat terug naar slaapmodus
5. ADC-conversie
6. De samplewaarde wordt uitgelezen door de LoRa-module



» Afbeelding 3.1, Uitlezen externe ADC [4]

Wanneer de GEN-node aangesloten en geconfigureerd wordt aan een EC5-sensor geeft dat het energieverbruik getoond in afbeelding 3.2. De gebeurtenissen zijn:

1. Reference wordt ingeschakeld en reference out (voeding voor de EC5) wordt ingeschakeld.
2. Conversietijd ADC. EC5 gebruikt 10mA vanaf reference out.
3. LoRa module verlaat sleepmodus en schakelt reference out uit.



» Afbeelding 3.2, Uitlezen applicatie EC5-sensor [4]

Hieraan is af te leiden dat het relatief veel stroom kost om de microcontroller actief te houden. Bij de EC5-sensoropstelling is er een aanzienlijk verbruik tijdens de ADC-conversie, te wijten aan de lange conversietijd. Het activeren van de ADC en de referentiespanning zorgen voor flinke pieken.

In een voorgaand project zijn er twee aanbevelingen gedaan om het energieverbruik te verbeteren. De eerste is de tijd beperken dat waarin de microcontroller actief is. In afbeelding 3.1 en 3.2 is te zien dat de het systeem er lang over doet om sensorgegevens op te halen. Om de microcontroller minder lang actief te laten zijn kan het ook dat de taken en activiteiten binnen de software verminderd worden. Dat zou mogelijk ten kosten kunnen gaan van functionaliteiten.

De tweede aanbeveling is het verlagen van de zogeheten oversampling ratio op de ADC. Hoe lager deze ratio, hoe hoger de datarate en dus ook de conversietijd. Bijkomend effect is wel, omdat er minder samples worden genomen, dat de nauwkeurigheid afneemt omdat de ruis toeneemt. [6]

Een lage ingangsstroom voor sensoren zoals de PT500 helpt ook bij het laag houden van het energieverbruik. De gain-mogelijkheden op de ADC maakt dat dit mogelijk is. Ook zou het proces veranderd kunnen worden waarbij er metingen opgespaard worden om in een latere



frequentie in 1 keer te sturen met LoRa. Ook zou het kunnen zijn dat de LoRa energiezuiniger kan worden geconfigureerd.

### **Samenvatting**

Hoe zou het energieverbruik verbeterd kunnen worden?

- Aanbeveling: Actieve tijd van de microcontroller beperken
  - In de testsoftware wordt de ADC een korte tijd in slaap gehouden voor het uitlezen van de sensoren. Deze tijd kan wellicht worden ingekort.
  - Taken of activiteiten schrappen of inkorten indien mogelijk
- Aanbeveling: sample rate verhogen van de ADC (oversampling)
- Kleine stroom voor de RTD (PT500)
- Overwegen minder vaak berichten te versturen (metingen bundelen)
- LoRa optimaliseren

## 4. Requirements

### 4.1 Functionele requirements

ID	Requirement	Prioriteit
R01	De firmware heeft driverondersteuning voor de ingangen Analog In, SPI en I2C en de uitgangen Analog Out en Digital Out.	Must
R02	De firmware heeft een implementatie voor het aansturen en het uitlezen van de externe ADC.	Must
R03	De firmware geeft de keuze voor het gebruik van analoge sensoren, zodat er ondersteuning is voor een sensoren met een uitgangssignaal van 4-20 mA, 0-2,5 V, 0-5 V en 0-10 V.	Must
R04	De firmware geeft de keuze voor het toepassen van de benodigde gain, zodat er ondersteuning is voor analoge sensoren met een zwak uitgangssignaal.	Must
R05	De firmware heeft ondersteuning voor geselecteerde driedraadssensoren (NO <sub>2</sub> en SO <sub>2</sub> ).	Must
R06	Uit de ondersteunde ingangssignalen van sensoren kan worden gekozen zonder dat hiervoor gewisseld hoeft te worden van firmware.	Should
R07	De firmware heeft de keuze een kleine uitgangsstroom in te schakelen van 0,5mA waarmee de PT500-sensor gevoed kan worden.	Should
R08	De firmware heeft generieke ondersteuning voor SPI- en I <sup>2</sup> C-sensoren.	Could

» Tabel 5.3, Functionele requirements

### 4.2 Niet-functionele requirements

ID	Requirement
R09	Het energieverbruik is lager dan het huidig verbruik wanneer de GEN-node wordt toegepast voor het uitlezen van analoge sensoren met de externe ADC.
R10	De firmware is modulair opgebouwd zodat de modules herbruikbaar zijn.
R11	De firmware is uitbreidbaar zodat er in de toekomst ook sensoren met andere communicatieprotocollen toegepast kunnen worden.
R12	De firmware is zodanig ontworpen dat het flexibiliteit biedt voor toekomstig gebruikte analoge sensoren

» Tabel 5.4, Niet-functionele requirements

## 5. Systeemeisen

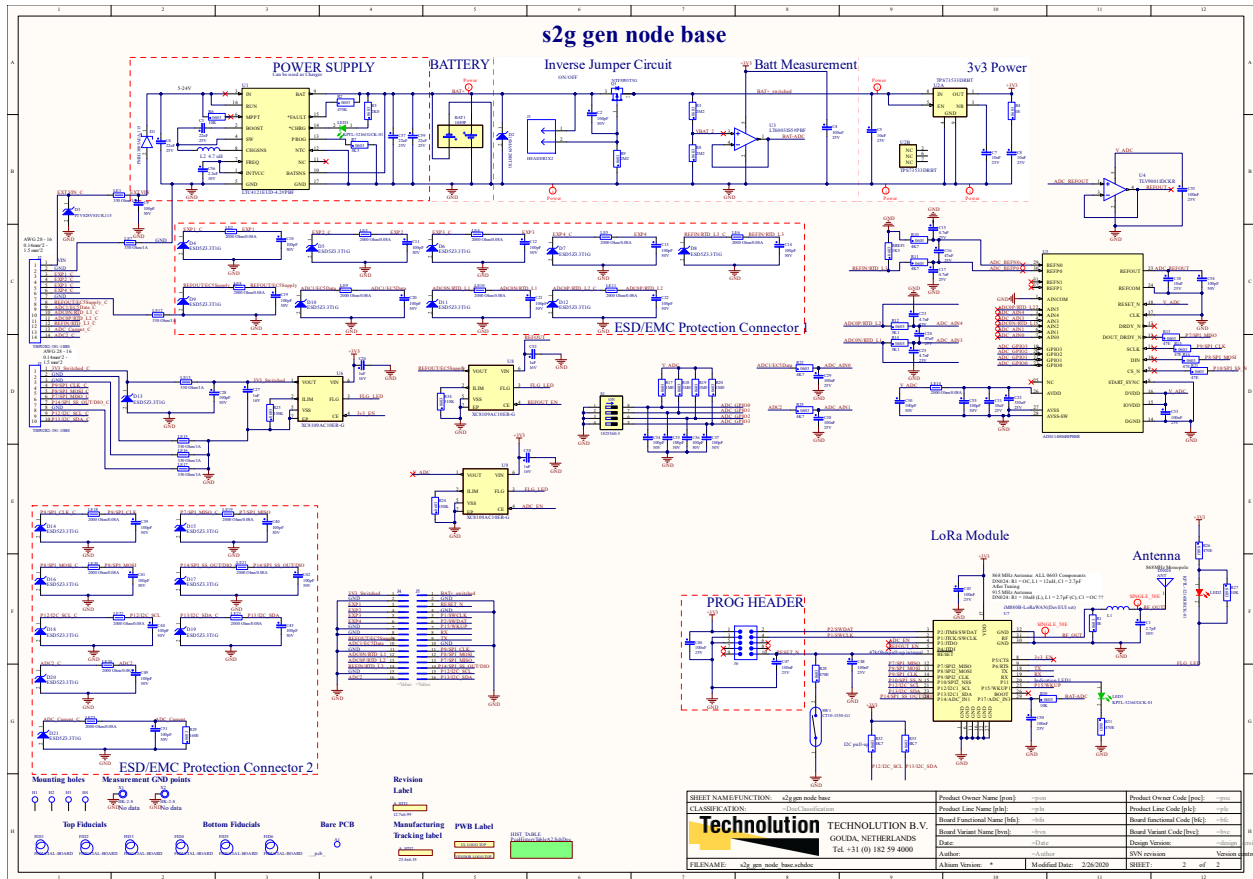
Omdat de firmware moet draaien op de hardware van de GEN-node zijn de systeemeisen afgestemd op deze node. De derde generatie is de meest recente node. De GEN-node bevat een microcontroller die een beperkt flash-geheugen en RAM-geheugen heeft. Daarom is het van belang dat de firmware aan de limieten van de hardware voldoet.

- GEN-node generatie 3 of nieuwer
- 128 kB Flash
- 32 kB RAM

# Literatuur

- [1] P. Ram, „LPWAN, LoRa, LoRaWAN and the Internet of Things,” Medium.com, 7 augustus 2018. [Online]. Available: <https://medium.com/coinmonks/lpwan-lora-lorawan-and-the-internet-of-things-aed7d5975d5d>. [Geopend februari 2020].
- [2] M. d. Rooij, „Internship Report GEN,” Gouda, 2018.
- [3] IMST GmbH, „Datasheet WiMOD iM880B,” Kamp-Lintfort, 2016.
- [4] N. Labordus, „Design Sense2Grow GEN node,” Technolution, Gouda, 2020.
- [5] Sparkfun, „Analog to Digital Conversion: Relating ADC Value to Voltage,” [Online]. Available: <https://learn.sparkfun.com/tutorials/analog-to-digital-conversion/relating-adc-value-to-voltage>. [Geopend 19 februari 2020].
- [6] Texas Instruments, „Datasheet ADS114S0xB,” 2017.
- [7] Technolution, „Sense and Grow - Basic design node software,” Gouda, 2016.
- [8] S. V. K. Krishnaswamy, Industrial Instrumentation, Delhi: New Age International, 2005.

# Bijlage 1 - Technische tekening GEN-node



Bijlage C

# Ontwerprapport

# Verbeterde firmware GEN-node



## Ontwerprapport

*Mei 2020, Gouda*

**Opdrachtgever**

Olaf Peters

**Auteur**

Jeremy Hut



DE HAAGSE  
HOGESCHOOL

# Inhoudsopgave

<b>1. Inleiding</b>	<b>2</b>
<b>2. Universele ondersteuning analoge sensoren</b>	<b>3</b>
2.1 Ondersteuning externe ADC	3
Overzichtsdiagram	3
Ontwerpkeuzes	4
Processtroom	6
Aansluiting	6
2.2 Ondersteuning alle gewenste sensoren	7
Overzichtsdiagram	7
Ontwerpkeuzes	7
Processtroom	8
Klassediagram	8
Sequentiediagram	10
Aansluiting	11
<b>3. Driedraadssensoren</b>	<b>13</b>
3.1 Alphasense sensoren	13
3.2 RTD	15
Aansluiting	15
<b>4. Energiezuinigheid</b>	<b>16</b>
4.1 Energie	16
Tijd beperken dat de microcontroller actief is	16
ADC energiezuiniger	17
LoRa	19
4.2 Verbeteringen drivers	20
SPI	20
I2C	21
<b>Literatuur</b>	<b>23</b>



# 1. Inleiding

In het ontwerprapport is de softwarearchitectuur van de derde generatie GEN-node te vinden. Dit is onderdeel van het Sense2Grow-project. Dit moet uiteindelijk leiden tot verbeterde firmware voor de GEN-node. Het ontwerp is opgedeeld in 3 iteraties en dus ook drie hoofdstukken, zoals opgesteld in het Plan van Aanpak.

Hoofdstuk 2 bevat het ontwerp dat antwoord geeft op de deelvraag: “Hoe kunnen analoge sensoren universeel worden ondersteund?”. Bij hoofdstuk 3 hoort de deelvraag: “Hoe kunnen geselecteerde sensoren ten behoeve van het meten van luchtkwaliteit worden ondersteund?”. En tot slot zal hoofdstuk 4 zich richten op het ontwerp voor de deelvraag: “Hoe kan het energieverbruik worden verlaagd?”.

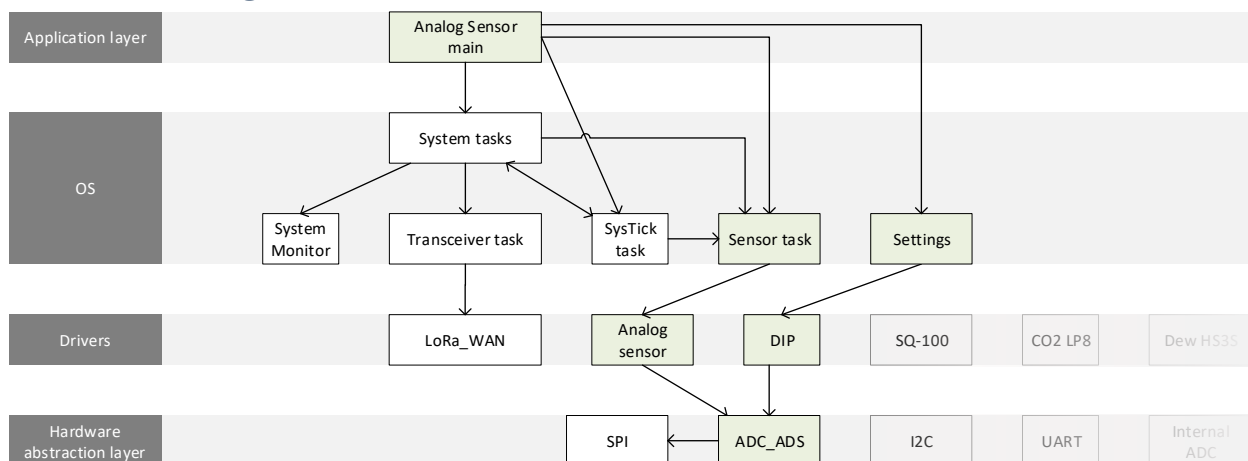
## 2. Universele ondersteuning analoge sensoren

Er is ondersteuning nodig voor veelgebruikte sensoren op de GEN-node. Dit ontwerp wordt in twee stappen gerealiseerd. De eerste stap is het ondersteunen van de analoge sensoren. De tweede stap is het kunnen kiezen voor het gebruik van andere gewenste sensoren zonder dat hiervoor gewisseld hoeft te worden van firmware.

### 2.1 Ondersteuning externe ADC

De ondersteuning van analoge sensoren op de externe ADC vereist een ontwerp dat een breed scala aan analoge sensoren ondersteunt. Voor een generieke oplossing is het nodig dat de firmware gelijk blijft bij het wisselen van een analoge sensor. In de analyse is naar voren gekomen dat hiervoor gewisseld moet kunnen worden tussen de versterkingsfactor.

#### Overzichtsdiagram



» Afbeelding 2.1, elementen en relaties analog\_sensor-applicatie

De blokken in afbeelding 2.1 in het groen zijn nieuw of gewijzigd ten opzichte van de al bestaande software.

Nieuwe of gewijzigde functies uit het diagram (grijs gemarkeerd):

- Analog Sensor main: laat system tasks, sensor task en systick task draaien
  - Main()
  - ApplicationStartup()
  - ApplicationSpecificInit()

- TaskSensorSetSamplesAggregateNr(int samples)
  - TaskSensorWakeUp()
  - TaskSensorInit()
- Sensor task (analog sensor): Zorgt ervoor dat de analoge sensor uitgelezen wordt met de juiste settings.
  - analog\_sensor\_TaskSensorInit()
  - analog\_sensor\_TaskSensor()
  - SendSensorMsg(float voltage)
  - SendSensorTimeOutMsg()
- Analog sensor: Leest analoge sensor uit op de ADC
  - AnalogSensorInit()
  - AnalogSensorDeInit()
  - AnalogSensorUpdateValue()
  - AnalogSensorGetValue()
- DIP: Lees de instellingen uit van de DIP-switches
  - DipSwitchInit()
  - DipSwitchDeInit()
  - DipSwitchUpdateValue()
  - DipSwitchGetValue()
- ADC\_ADS114S0B: Aansturen van de specifieke ADC aan de hand van SPI.
  - AdcExternalInit()
  - AdcExternalDeInit()
  - AdcExternalReadData()
  - AdcExternalReadPort()
  - Adc\_ADS114S0B\_WriteRegister()
  - Adc\_ADS114S0B\_ReadRegister()
  - Adc\_ADS114S0B\_Start()
  - Adc\_ADS114S0B\_...

## Ontwerpkeuzes

### Versterking instellen: DIP-switches

Het is nu niet bekend of er sprake is van sensor die versterking (gain) nodig heeft, een sensor waarvan de spanningswaarde omgerekend moet worden naar een andere eenheid (zoals graden Celcius in het geval van temperatuur) en of de sensor de spanning, stroom of weerstand verandert. Voor het lezen van het voltage is in de software de versterking de belangrijkste variabele. Om dit extern in te kunnen laten stellen zou deze instelling ontvangen kunnen worden via LoRa. Op de PCB zijn ook DIP-switches beschikbaar. Deze zijn

uitleesbaar via de externe ADC. De meest gebruiksvriendelijke oplossing is om de versterking op de DIP-switches in te laten stellen door de eindgebruiker. Als alle versterkingswaarden instelbaar zijn gebruikt dit 8 van de 16 mogelijkheden. Dit geeft als resultaat deze mogelijkheden:

DIP-setting	Functie	DIP-setting	Functie
0000	gain 1	1000	vrij
0001	gain 2	1001	vrij
0010	gain 4	1010	vrij
0011	gain 8	1011	vrij
0100	gain 16	1100	vrij
0101	gain 32	1101	vrij
0110	gain 64	1110	vrij
0111	gain 128	1111	vrij

» Tabel 2.1, instellingsmogelijkheden DIP-switch (applicatie: analog sensor)

### Settings-module

Deze module haalt de settings op die nodig zijn voor het uitlezen van de aangesloten sensor. Deze is afhankelijk van de DIP-switch maar zou voor het ophalen van de instellingen ook afhankelijk gemaakt kunnen worden van andere communicatiemiddelen, zoals LoRa. De settings worden alleen opgehaald bij het starten van de GEN. Dit bespaart energie in vergelijking met wanneer dit bij elke sensormeting wordt aangeroepen. Als een sensor wordt gewisseld moet de GEN gereset worden, dit kan met een magneet.

### Voltages opsturen, berekeningen extern

Het versturen van de metingen naar de backoffice kan gedaan worden in voltages. Omdat de software generiek is, is de juiste eenheid is die hoort bij de sensor en hoe dit omgerekend moet worden onbekend. De sensornode zou deze gegevens wel kunnen ontvangen via LoRa bij het opstarten. Een alternatieve oplossing is dat deze berekening opgeslagen is in de backend en de berekeningen hier ook gedaan wordt. Het voordeel hiervan is dat de GEN-node minder berekeningen hoeft te doen, wat voordelig is voor dit soort low power apparaten.

### Abstractie - analoge sensor

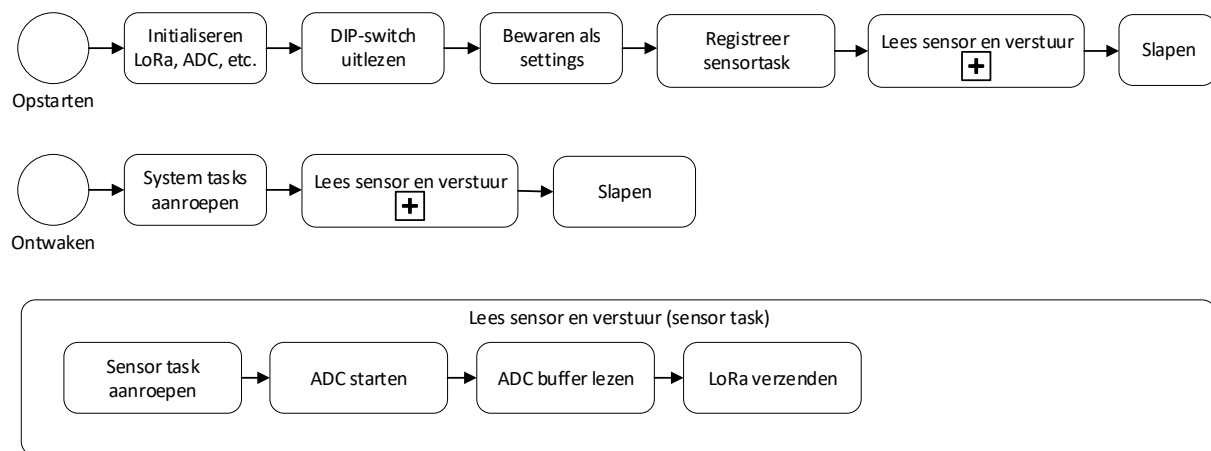
In het geval het nodig is dat er in de toekomst van een ander model ADC gebruik moet worden gemaakt is er gekozen voor een abstracte interface op deze specifieke ADC. In het rijtje van sensoren valt deze interface onder 'analoge sensor'. De implementatie bevindt zich in de hardware abstraction layer. Hier staat de specifieke aansturing voor dit model ADC.

## Abstractie - DIP-switch

Voor de DIP-switch is dezelfde keuze gemaakt als de analoge sensor. Dit geeft flexibiliteit wanneer er voor andere hardware gekozen wordt. Er kan in de toekomst ook gekozen worden om een uitbreiding of vervanging te doen met LoRa, waarbij er via een LoRa-bericht de juiste instellingen wordt gestuurd vanuit de backoffice.

## Processtroom

In afbeelding 2.2 is te zien dat de DIP-switch alleen uitgelezen wordt tijdens het opstarten van de GEN-node. Dat betekent dat bij het veranderen van de DIP-switch, de GEN-node gereset moet worden voordat de nieuwe instellingen worden doorgevoerd.



» Afbeelding 2.2, Processtroom van de applicatie

## Aansluiting

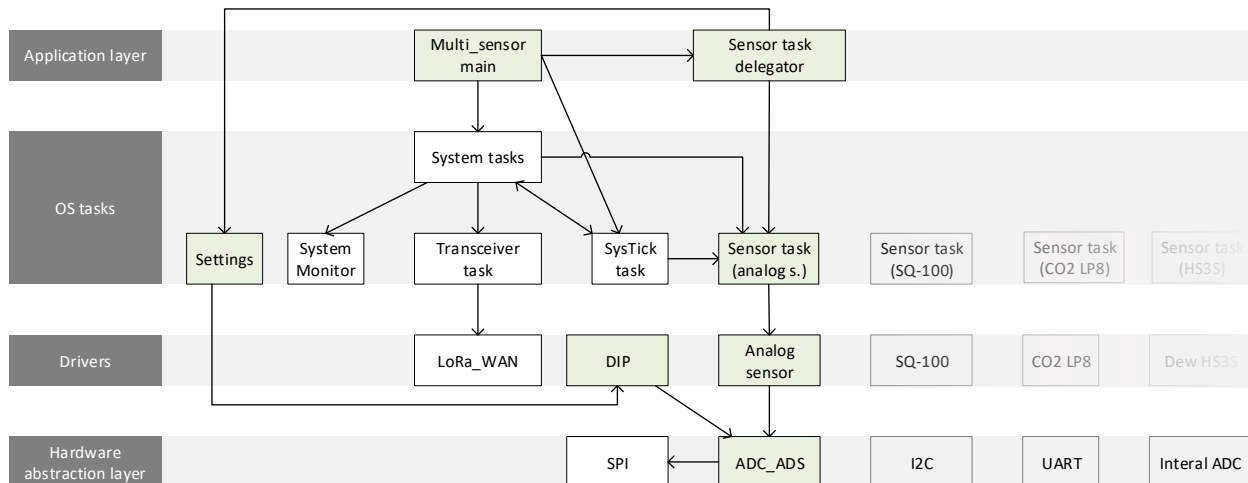
Een analoge sensor kan aangesloten worden op de volgende poorten.

- 3V3\_OUT (3,3V voeding)
- Gnd
- ADC2

## 2.2 Ondersteuning alle gewenste sensoren

Voor het ondersteunen van overige sensoren op dezelfde firmware is het volgende ontwerp gemaakt. Dit is een aanvulling op het ontwerp van hoofdstuk 2.1.

### Overzichtsdiagram



» Afbeelding 2.3, elementen en relaties multi\_sensor-applicatie

In dit klassendiagram wordt er getoond wat er gebeurt als er gekozen is voor een analoge sensor. Wanneer bijvoorbeeld voor een LP8-sensor gekozen wordt zal er een andere sensor task uitgevoerd worden die gebruik maakt van de juiste driver (LP8) en HAL-laag (I<sup>2</sup>C).

### Ontwerpkeuzes

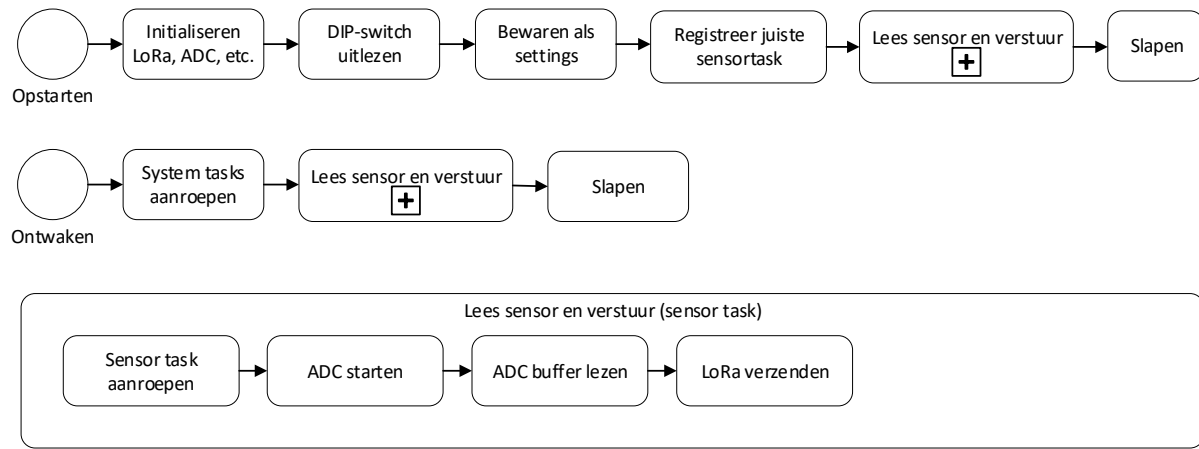
- Sensor task delegator
  - Er bestaan verschillende sensor tasks. Sensor task delegator registreert de juiste sensor task aan het OS op basis van de 'settings'.
- DIP
  - Hiermee wordt gekozen welke sensor er toegepast moet worden. Links de DIP-setting waarmee analoge sensoren aangesloten kunnen worden. Rechts de lijst met niet-analoge sensoren die ook ingesteld kunnen worden.

DIP-setting	Sensor	DIP-setting	Sensor
0000	An - Gain 1	1000	3-wire gain 1 (NO2/SO2)
0001	An - gain 2	1001	3-wire gain 8 (PT500)
0010	An - gain 4	1010	Ds18b20
0011	An - gain 8	1011	Sh31 Celcius
0100	An - gain 16	1100	LP8 CO2
0101	An - gain 32	1101	Hs3s
0110	An - gain 64	1110	SPS30

0111	An - gain 128	1111	vrij
------	---------------	------	------

» Tabel 2.2, instellingsmogelijkheden multi\_sensor-applicatie op DIP-switches

## Processtroom

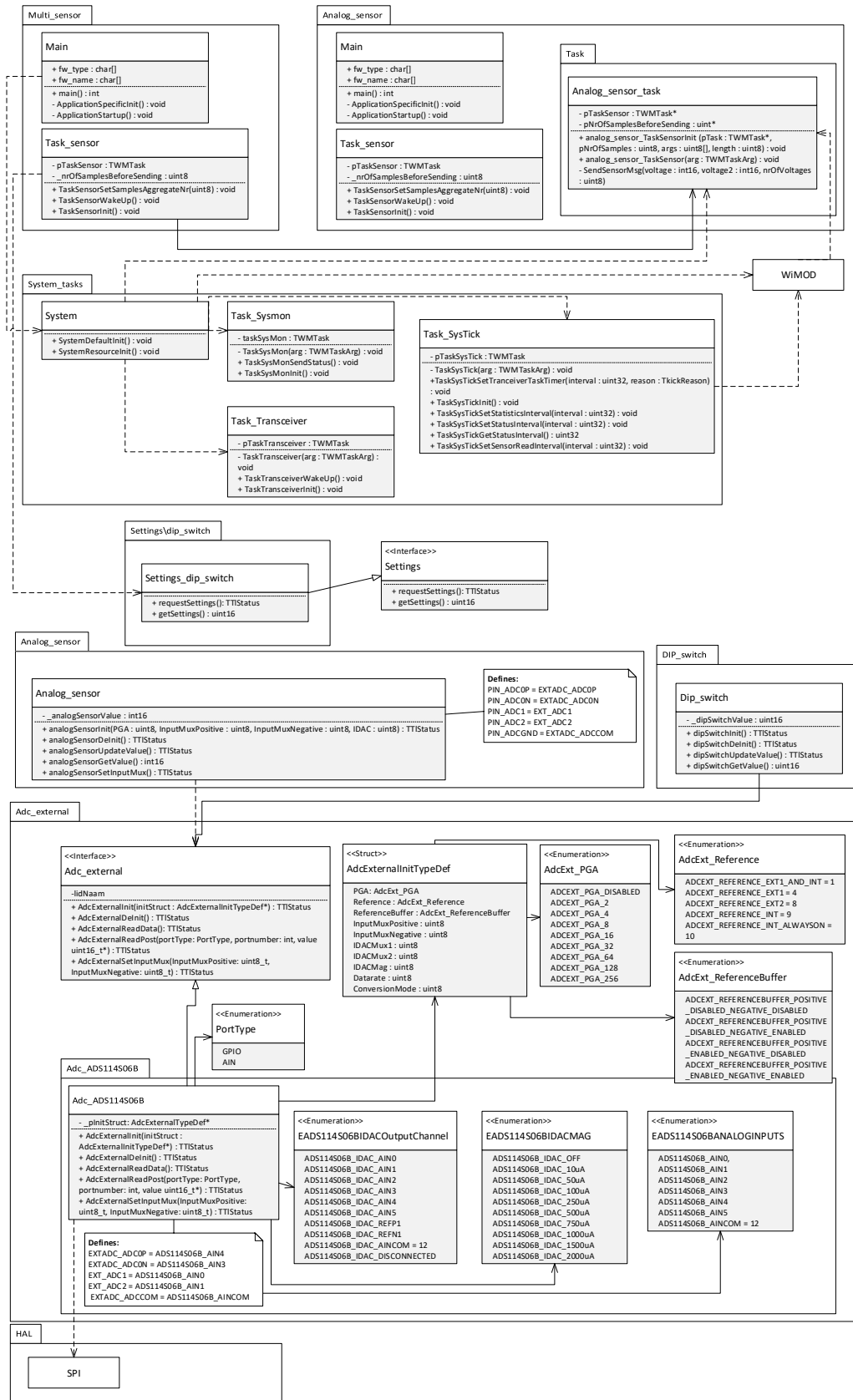


» Afbeelding 2.4, Processtroom

Zoals te zien in afbeelding 2.4 is het sequentiediagram niet veel veranderd ten opzichte van het diagram voor de analog\_sensor-applicatie. Het verschil is dat het registreren van de juiste sensortask wat omvangrijker is, aangezien dit nu afhankelijk is van de settings-module (DIP-switch).

## Klassediagram

Zie afbeelding 2.5 voor een klassediagram van de multi\_sensor-applicatie die gebruik maakt van de sensortask van de analog\_sensor-applicatie. Een pakket staat voor een module en een klasse voor een .c/.h-bestand

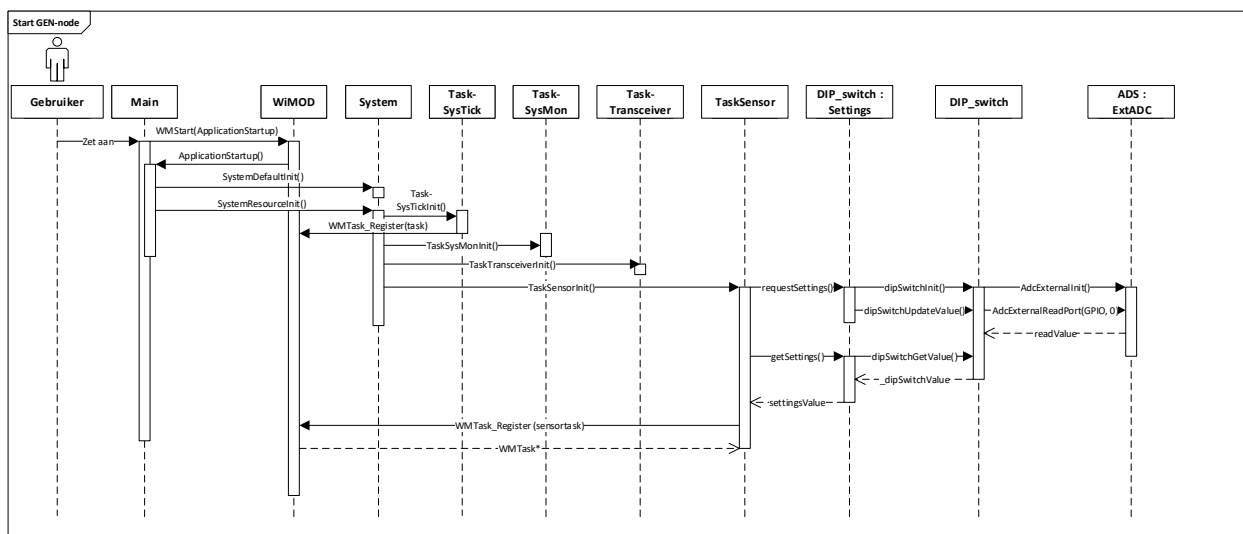


» Afbeelding 2.5, klassediagram

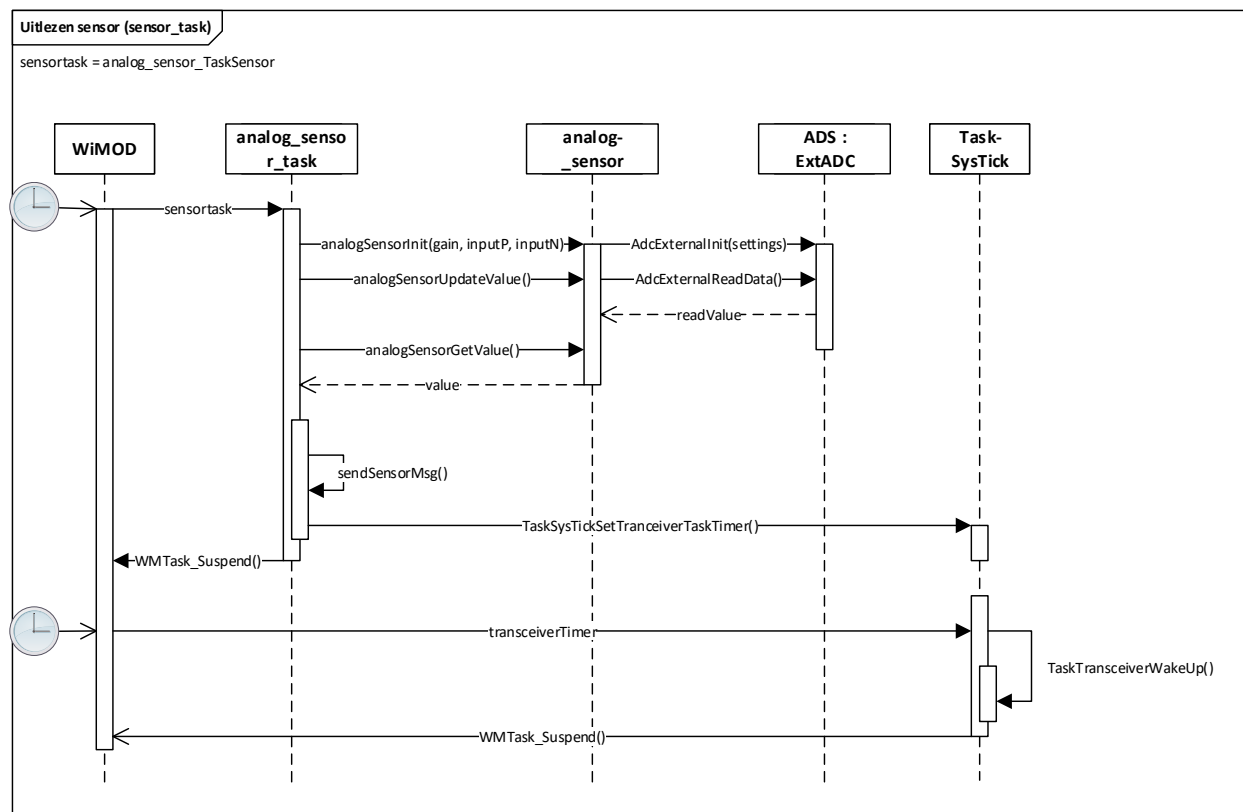


## Sequentiedigram

Deze sequentiediagrammen in afbeelding 2.6 en 2.7 zijn een detailontwerp van het geschetste proces in afbeelding 2.4.



» Afbeelding 2.6, sequentiediagram opstarten

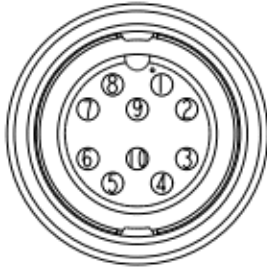


» Afbeelding 2.7, sequentiediagram uitlezen

## Aansluiting

De sensoren kunnen op de volgende poorten worden aangesloten op de GEN-node:

### Ds18b20



- 1: 3v3
- 2: Gnd
- 8: SDA

### Sh31

---

			7			13
			8			14

- 7: SDA
- 8: SCL (I2C)
- 13: Gnd
- 14: 3v3

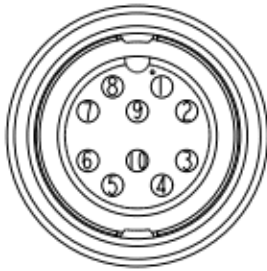
### LP8

---

13			7			1
14			8			2

- 1: Rx
- 2: Tx
- 7: SDA
- 8: SCL (I2C)
- 13: Gnd
- 14: 3v3

### Hs3s



- 1: 3v3 rood
- 2: Gnd zwart
- 7: SCL (I2C) oranje
- 8: SDA bruin

### SPS30

In waterbestendige doos.

- 1: 3v3
- Groen/Geel: Gnd
- 2: SCL (I2C)
- 3: SDA

## 3. Driedraadssensoren

Dit hoofdstuk gaat over het ontwerp van de ondersteuning van NO<sub>2</sub>- en SO<sub>2</sub>-sensoren. Dit zijn driedraadssensoren. Ook wordt de PT500 (RTD) meegenomen.

### 3.1 Alphasense sensoren

#### Software-applicatie

De software is een uitbreiding van de analog\_sensor-applicatie. Door bij de initialisatie van de bijbehorende taak een modusnummer mee te geven kunnen de bijbehorende instellingen voor de ADC worden uitgevoerd.

#### Sensor

De NO<sub>2</sub>/SO<sub>2</sub>-sensoren hebben een uitgangsspanning tussen 0 – 5V. Dat zal teruggebracht worden naar 0 – 2.5V. De PT500 heeft een ingangsstroom nodig van 0,5 mA en een gain van 8. Hierbij zal de maximale uitgangsspanning dan 2,5 V zijn.

#### Sensor: schakeling en behuizing

De NO<sub>2</sub>-B43F en de SO<sub>2</sub>-B4 zijn aangesloten op een PCB van de fabrikant: de Alphasense ISB. Dit bord wordt gevoed door een converter die 3,3V opschaaft naar 5V. Dit is gemonteerd in een waterbestendige doos met 4 draden.



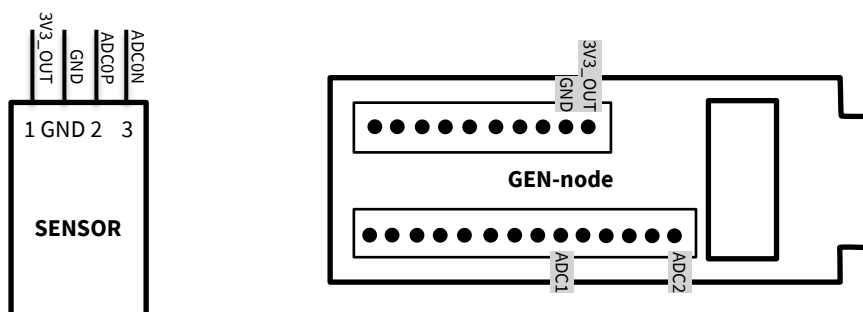
Afbeelding 3.1, Aansluitingen Alphasense Individual Sensor Board (ISB) [1]

Voeding (5V): VIN+ draad 1, VIN- ground

Werkende elektrode: OP1+ draad 2

Hulpelektrode: OP2+ draad 3

#### Sensor aansluiten op GEN-node



### Aanvullende instellingen NO2/SO2

- 2x meten: AIN1 (ADC2) en AIN0 (ADC1)
- Gain: 1
- 3v3 out aanlaten of alleen tijdens uitlezen aan met voldoende opstarttijd (>65 seconden)

### Bericht

- 0x0A: Analog In (mV) (uint16)
- 0X10: Auxillary Electrode (mV) (uint16) (nieuw type measurement)

### Formule

De omrekenformule is hieronder te vinden: [2]

$$NO_2 [ppb] = \frac{(V_{WE} - V_{WE,zero}) - n(V_{AE} - V_{AE,zero})}{Sensitivity}$$

Hierbij zijn de spanningen in mV.  $V_{WE,zero}$ ,  $V_{AE,zero}$  en  $Sensitivity$  zijn te vinden in de meegeleverde documenten vanuit de fabriek, dit is voor elke sensor anders.  $n$  is de factor afhankelijk van de temperatuur, deze is te vinden in de tabel van bovenstaande bron.

### Kalibratie

Door elektronische afwijkingen kan het nodig zijn de nulafwijking (zero) te kalibreren. Dat gebeurt aan de hand van deze vergelijking: [2]

$$(V_{WE} - V_{PCBWE}) - n(V_{AE} - V_{PCBAE}) = 0$$

Hierbij is  $V_{PCBWE}$  en  $V_{PCBAE}$  de elektronische nulafwijking op de ADC.

Naast temperatuur is de NO2-B43F ook gevoelig voor O<sub>3</sub>. Volgens het RIVM moet daarom de O<sub>3</sub>-waarde ook meegenomen worden in de kalibratie [5]. Daarnaast is de sensor ook gevoelig voor de luchtvochtigheid. In de ideale situatie wordt daarmee dus ook gekalibreerd [3][4]. De sensor gaat één tot twee jaar mee, maar zal na enkele maanden al in gevoeligheid afnemen. Maandelijks herkalibreren wordt geadviseerd [6].

Voor de SO2-B4 is een kalibratie afhankelijk van de temperatuur en de nulafwijkingen voldoende. [7]

## 3.2 RTD

Voor de PT500 sensoren is er een kleine stroom nodig. In het voorgaande project is geadviseerd dit op 0,5mA te zetten. [10]

De PT500 heeft een weerstand van 460-620  $\Omega$  in normale omstandigheden. Dat betekent een uitgang tussen de 0,23 V en 0,31 V met 0,5 mA (wet van Ohm). Dit betekent een versterkingsfactor van 8x (of 4x bij extreme temperaturen). Daarnaast is het een driedraadssensor, alleen hoeft hierbij niet 2x gemeten te worden aangezien beide te meten polen dezelfde versterkingsfactor hebben. Het gebruik van ADC0P en ADC0N als input is voldoende. Het resultaat is het verschil tussen de draadsweerstand en de sensor inclusief de draadsweerstand.

Door wederom gebruik te maken van de task analog\_sensor en het toepassen van een nieuwe modus kan ook de RTD zoals de PT500 worden gebruikt. De modus zal dus de versterking op 8 zetten en de positieve inputMux op ADC0P en negatieve inputMux op ADC0N. Tot slot zal de IDAC geactiveerd moeten worden met een stroom van 0,5mA vanaf de externe ADC.

### Aansluiting

De RTD bestaat uit 3 draden:

- L3: REFIN
- L2: ADC0P
- L1: ADC0N

## 4. Energiezuinigheid

Onder deze iteratie vallen verbeteringen aan de energiezuinigheid en verbeteringen aan drivers.

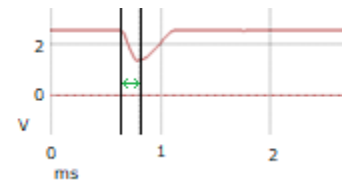
### 4.1 Energie

Het ontwerp valt samen met aanvullend onderzoek aan de hand van stroommetingen. Dit onderzoek is gebaseerd op aanbevelingen die gedaan zijn bij het vorige project.

#### Tijd beperken dat de microcontroller actief is

##### Terugkeren naar het OS

Volgens mijn begeleider zou het veel tijd kosten om tijdens het uitvoeren van een taak steeds terug te gaan naar het OS. Momenteel wordt een taak herhaaldelijk aangeroepen vanuit het OS totdat deze klaar is omdat er anders een watchdog timer overflow gebeurt, wat het systeem reset.



» Afbeelding 4.1, Tijd dat het terugkeren naar het OS kost

Uit metingen blijkt dat de tijd dat het kost 0,1 ms is, zie afbeelding 4.1. Door een eenvoudige applicatie te maken waarbij een pin (rode lijn) omlaag wordt gebracht wanneer de taak terugkeert naar het OS en de pin omhoog te brengen bij het terugkeren naar de taak wordt de tijd die hiertussen zit inzichtelijk.

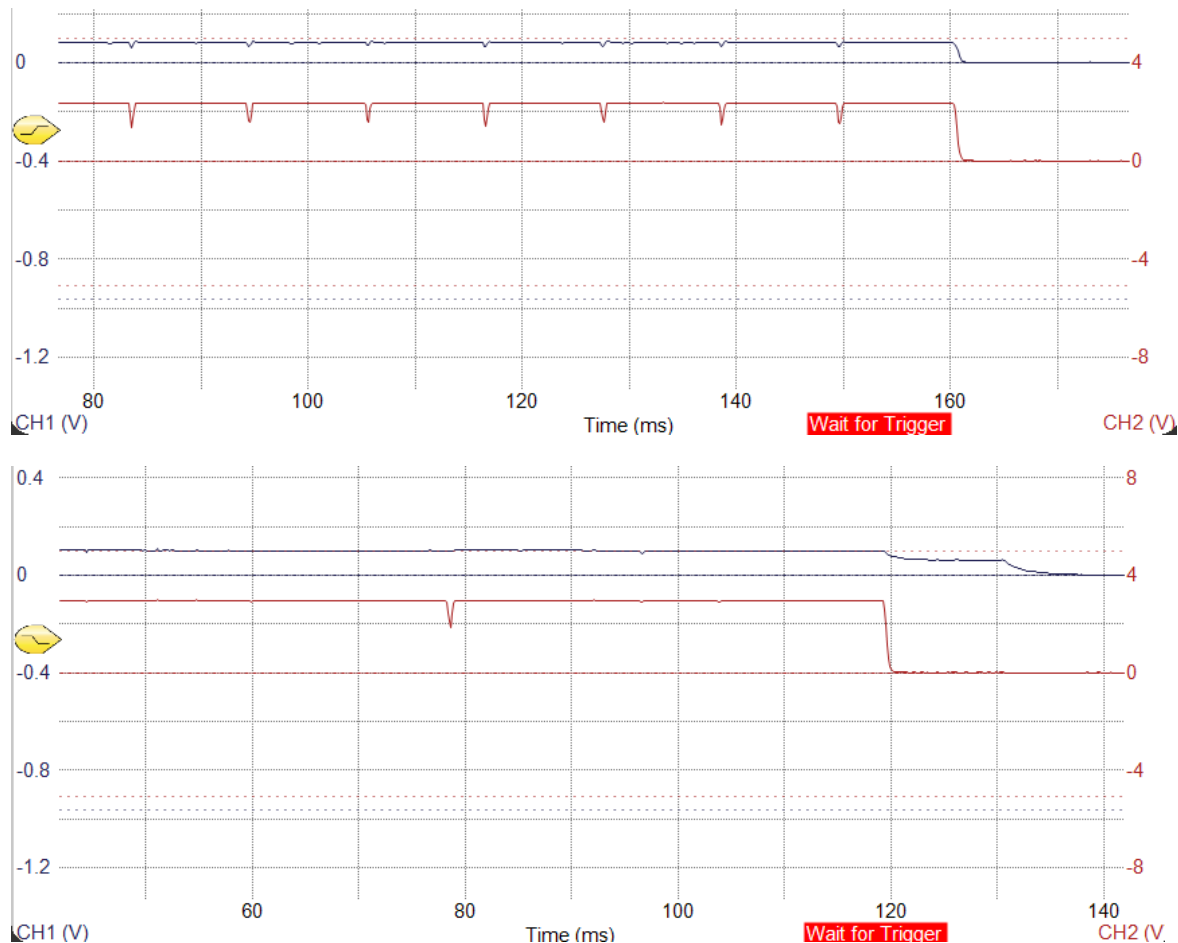
Het alternatief is om de software zodanig aan te passen dat er niet tijdens het uitvoeren van een taak wordt teruggekeerd naar het OS. Dit kan door een met enige regelmaat een watchdog trigger uit te voeren, zodat de watchdog timer niet overloopt. Het uitvoeren van deze trigger blijkt uit metingen eveneens 0,1ms te kosten. De conclusie is dat hierin geen winst valt te behalen, in tegenstelling tot wat voorheen werd aangenomen.

##### Logging

Eén van de functionaliteiten die processortijd kost, terwijl het in praktijk alleen gebruikt wordt voor testomgevingen, is de UART-log. Ik kwam hierachter door stroom te meten met en zonder log. De applicatie voor deze test bestond voornamelijk herhaaldelijk uit een slaap van 10ms en het wel of niet printen van 4 berichten van 83 karakters. Zonder log kost het om dit te doorlopen 11ms. Wordt het printen van de 4 logberichten ingeschakeld zorgt dit voor een

doorlooptijd van 40ms. De rode lijn toont het actief zijn van een taak aan, en de blauwe lijn het stroomverbruik. Opvallend is dat er in de tweede situatie na het stoppen van de taak een uitloop is van 12ms.

Naast het feit dat logging bijna tien kilobyte aan flash-geheugen kost, houdt logging de microcontroller dus ook langer actief. Omdat in productieomgeving deze logging niet uitgelezen wordt heb ik ervoor gekozen om dit uit te schakelen. Deze moet optioneel in te schakelen zijn voor foutoplossing.



» Afbeelding 4.2, Energieverbruik met logging (boven) en zonder logging (onder).  
Blauwe lijn: stroomverbruik, rode lijn: taak actief (hoog) of inactief (laag)

## ADC energiezuiniger

### Samplerate

Het uitlezen van de analoge sensor met de ADC gaat met een bepaalde samplerate (datarate). In het vorige project is deze op 20 SPS gezet. Hierbij kwam de aanbeveling om te onderzoeken of dit verhoogd kan worden. Een hogere samplerate zorgt namelijk voor meer ruis, maar ook zorgt het voor een kortere tijd die de ADC nodig heeft om een meting te doen.



**Table 2. Effective Resolution From RMS Noise (Noise-Free Resolution From Peak-to-Peak Noise)  
at AVDD = 3.3 V, AVSS = 0 V, PGA Enabled, and Internal 2.5-V Reference**

DATA RATE (SPS)	GAIN							
	1	2	4	8	16	32	64	128
2.5	16 (16)	16 (16)	16 (16)	16 (16)	16 (16)	16 (16)	16 (16)	16 (16)
5	16 (16)	16 (16)	16 (16)	16 (16)	16 (16)	16 (16)	16 (16)	16 (16)
10	16 (16)	16 (16)	16 (16)	16 (16)	16 (16)	16 (16)	16 (16)	16 (16)
16.6	16 (16)	16 (16)	16 (16)	16 (16)	16 (16)	16 (16)	16 (16)	16 (16)
20	16 (16)	16 (16)	16 (16)	16 (16)	16 (16)	16 (16)	16 (16)	16 (16)
50	16 (16)	16 (16)	16 (16)	16 (16)	16 (16)	16 (16)	16 (16)	16 (15.5)
60	16 (16)	16 (16)	16 (16)	16 (16)	16 (16)	16 (16)	16 (16)	16 (15.4)
100	16 (16)	16 (16)	16 (16)	16 (16)	16 (16)	16 (16)	16 (15.8)	16 (14.9)
200	16 (16)	16 (16)	16 (16)	16 (16)	16 (16)	16 (16)	16 (15.3)	16 (14.5)
400	16 (16)	16 (16)	16 (16)	16 (16)	16 (16)	16 (15.8)	16 (14.7)	16 (14.0)
800	16 (16)	16 (16)	16 (16)	16 (16)	16 (16)	16 (15.2)	16 (14.3)	16 (13.6)
1000	16 (16)	16 (16)	16 (16)	16 (16)	16 (15.8)	16 (14.9)	16 (14.2)	16 (13.3)
2000	16 (15.9)	16 (14.9)	16 (15.3)	16 (15.2)	16 (14.8)	16 (14.5)	16 (13.5)	15.2 (12.6)
4000	16 (13.0)	16 (12.6)	15.7 (12.9)	16 (13.1)	15.6 (13.0)	15.5 (12.9)	14.4 (11.9)	13.6 (10.9)

» Tabel 4.1, effectieve resolutie bij verschillende datarates

In bovenstaande tabel is te vinden bij welke datarate en welke versterkingsfactor de effectieve resolutie is. Een resolutie van 16 is het maximum. Alles wat hieronder zit heeft vanwege de ruis een negatief effect op de nauwkeurigheid.

NOMINAL DATA RATE <sup>(1)</sup> (SPS)	FIRST DATA FOR CONTINUOUS CONVERSION MODE OR SINGLE-SHOT CONVERSION MODE <sup>(2)</sup>	
	ms	NUMBER OF t <sub>MOD</sub> PERIODS
2.5	406.559	104079
5	206.559	52879
10	106.559	27279
16.6	60.309	15439
20	56.559	14479
50	20.211	5174
60	16.965	4343
100	10.211	2614
200	5.211	1334
400	2.711	694
800	1.461	374
1000	1.211	310
2000	0.711	182
4000	0.461	118

» Tabel 4.2, Tijd dat de ADC nodig heeft voor het doen van een meting

In bovenstaande tabel staat de tijd in milliseconden die de ADC nodig heeft om een meting te doen (conversie). Dit is afhankelijk van de datarate. Een gulden middenweg is een datarate van 400 SPS omdat dit zorgt voor een erg beperkte conversietijd (3ms), terwijl het voor de verschillende versterkingsfactoren de ruis geen of weinig invloed heeft.

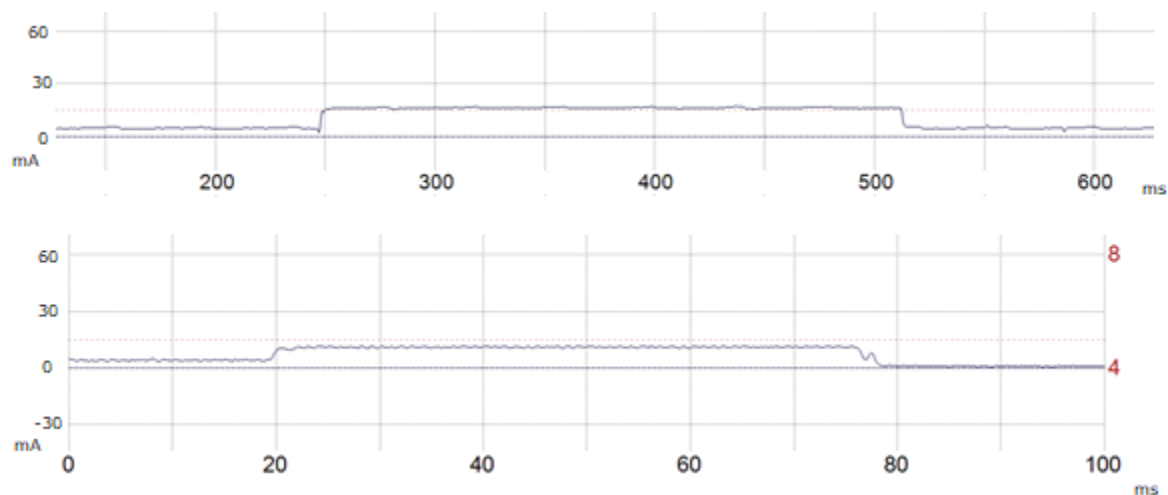
## Analog sensor applicatie

De microcontroller kan tijdens het uitvoeren van de conversie in een energiezuinige slaap vallen. Dit is ook gebruikt tijdens het uitvoeren van de metingen in het vorige project, waarbij ook een lage datarate werd toegepast (20SPS, 57ms conversietijd). Hierbij moet wel de ADC actief blijven, evenals de voeding van de sensor. Ik heb ervoor gekozen om de microcontroller actief te houden gedurende de hele conversie, maar gebruik te maken van een kleine conversietijd (400SPS, 3ms). Dit voorkomt namelijk stroomverbruik dat nodig is om alles weer actief. Daarnaast wordt de ADC en de aangesloten sensor bijna 20x korter gevoed tijdens de conversie.

## LoRa

Een grootverbruiker van energie is het verzenden van LoRa-berichten. Helaas is hiervan weinig te beïnvloeden. Een deel van de aansturing bevindt zich een de statische bibliotheek van de LoRa-modulefabrikant. Een interessante instelling die wel instelbaar is, is de TxPower (zendvermogen). Origineel staat deze op 16 dBm. Als deze bijvoorbeeld op 4 wordt gezet, gaat de stroom tijdens het zenden met 40% omlaag. Een lagere TxPower betekent dat er voor langere afstanden een hogere spreading factor gekozen moet worden.

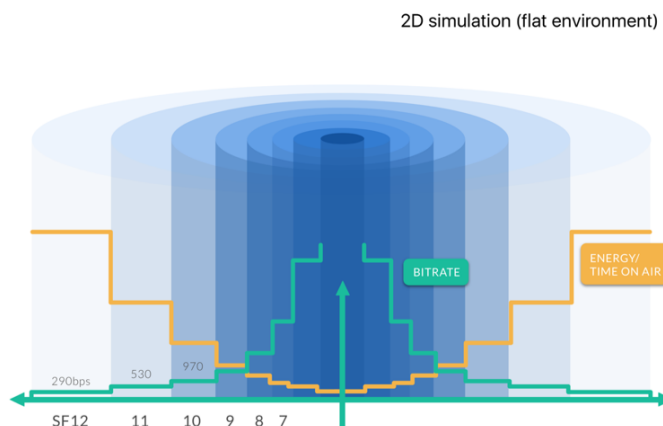
Met 16 dBm is de oppervlakte van de verzending van een bericht te berekenen door  $15 \text{ mA} \times \text{airtime}$ , zoals te zien is in afbeelding 4.4.



» Afbeelding 4.3, Stroommeting van verzending. Airtime boven: 270 ms, beneden: 56 ms

De airtime die nodig is voor het versturen van het bericht heeft veel invloed op het energieverbruik. De airtime is afhankelijk van drie factoren [8]: byte-lengte van het bericht, spreading factor (SF) en de bandbreedte. Op die laatste twee hebben we niet direct invloed in de firmware. De bandbreedte is in principe altijd 125 kHz. Alleen bij SF7 kan deze ook 250 kHz

zijn. Aangezien er gebruik gemaakt wordt van Adaptive Data Rate (ADR), wordt de spreading factor en bandbreedte voor de node dynamisch bepaald.



» Afbeelding 4.3, bron: <https://www.thethingsnetwork.org/article/how-spreading-factor-affects-lorawan-device-battery-life>

Aangezien er geldt dat hoe hoger de SF-waarde is, hoe langer de airtime is [9], wordt er dus een langere tijd minder stroom verbruikt bij een vermindering van de TxPower. Wanneer de SF-waarde met 1 verhoogd wordt betekent dit een verdubbeling voor de airtime en kan zo bij een slecht bereik veranderen van SF7 naar SF12. Dat is vergroting van  $2^5$  (32x). Dat weegt niet op tegen een iets lager stroomniveau. Het beste is dus om een middenweg te vinden voor de TxPower en SF-waarde, waarbij de SF-waarde vaak zo laag mogelijk kan zijn (SF7). SF7 lukt regelmatig met een TxPower van 16 dBm. Daarom is de keuze om de TxPower op 16 dBm te laten.

Metingen samenvoegen om in 1x te versturen scheelt wel in energieverbruik. Een 2x zo lang bericht (byte-length) betekent een airtime die zo'n 1,3x zo groot is. Hiermee valt dus winst te halen, maar dat betekent wel dat metingen meer vertraagd worden voordat ze zichtbaar worden in de backoffice. Dit hangt af van de klant en bijhorende toepassing. Daarom is ervoor gekozen het bundelen van metingen niet standaard toe te voegen in de firmware. Wel is het aantal metingen die opgespaard moeten worden voor verzending in de broncode eenvoudig aanpasbaar.

## 4.2 Verbeteringen drivers

### SPI

De SPI-driver bestaat uit te veel functies met hetzelfde doel. Deze driver wordt met dit ontwerp eenvoudiger en kleiner in omvang.

### Onaangepaste functies:

- `void SPI_DeInitialize()`
- `TTlStatus SPI_Transmit( uint8_t *data, uint32_t len, uint8_t *receive_buf )`

### Aangepaste functies:

- `void SPI_Init()`
  - SPI\_BaudRatePrescaler van 16 naar 8 voor een snellere communicatie (4MHz)
- `TTlStatus SPI_Transmit_Byte( uint8_t data, uint8_t *recv)`
  - Accepteert nu de parameter \*recv. De SPI receivewaarde moest al worden uitgelezen maar werd voorheen niet teruggegeven. Werkt als SPI\_Transmit maar vereist geen data-array om te verzenden. Roept in het nieuwe algoritme alleen SPI\_Transmit aan.

### Overbodige functies:

- `TTlStatus SPI_Transfer_two_bytes( uint8_t Sendata, uint8_t* received_data )`
  - Te specifieke variatie op SPI\_Transmit en vervangbaar door SPI\_Transmit
- `TTlStatus SPI_Transfer( uint16_t Sendata, uint8_t data_second_cycle, uint8_t* received_data )`
  - Te specifieke variatie op SPI\_Transmit en vervangbaar door SPI\_Transmit
- `void SPI_Send(uint16_t SenData )`
  - Nooit getest en overbodig
- `uint16_t SPI_Read(void)`
  - Nooit getest en overbodig

### Klasse

SPI
<div>+ SPI_Init() : void + SPI_Transmit(data : uint8 *, len : uint32, receive_buf : uint8 *) : TTlStatus + SPI_Transmit_Byte(data : uint8, receive_buf : uint8 *) : TTlStatus + SPI_DeInitialize() : void</div>

» Afbeelding 4.4, Nieuw ontwerp SPI-driver

## I2C

I2C-driver bevat functies die deels hetzelfde algoritme uitvoeren.

### Onaangepaste functies:

- `void HalI2cInit(uint32_t clockSpeed)`
- `TTIStatus HalI2cSendData(uint8_t address, uint8_t data)`
- `TTIStatus HalI2cSendDataBuffer(uint8_t address, uint8_t * data, uint8_t len)`
- `TTIStatus HalI2cRecvData(uint8_t address, uint8_t *data, uint8_t nbrOfBytes)`
- `void ResetStatesI2c()`
- `void HalI2cDeinit()`

Nieuwe functie:

- `static TTIStatus HalI2cSendAddress(uint8_t address)`

### Aangepaste functie:

- `TTIStatus HalI2cSendReceive(uint8_t i2cAddress, uint8_t * txData, uint8_t txLen, uint8_t * rxData, uint8_t rxLen)`
  - Het algoritme van deze functie bestaat uit grote delen van `Hall2cSendDataBuffer` en `Hall2cRecvData`. Het nieuwe algoritme van deze functie moet `Hall2cSendDataBuffer` en `Hall2cRecvData` aanroepen.

### Klasse

I2C
<div>+ HalI2cInit(clockSpeed : uint32) : void + HalI2cSendData(address : uint8*, data : uint8) : TTIStatus + HalI2cSendDataBuffer(address : uint8*, data : uint8*, len : uint8) : TTIStatus + HalI2cSendReceive(address : uint8, txData : uint8*, txLen : uint8, rxData : uint8*, rxLen : uint8) : TTIStatus + HalI2cRecvData(address : uint8, data : uint8*, nbrOfBytes : uint8) : TTIStatus + ResetStatesI2c() : void + HalI2cDeinit() : void</div>

» Afbeelding 4.5, Nieuw ontwerp I2C-driver

# Literatuur

1. "Datasheet Individual Sensor Board (ISB)", Alphasense,  
<http://www.alphasense.com/WEB1213/wp-content/uploads/2019/10/ISB.pdf>
2. "Correcting for background currents in four electrode toxic gas sensors", Alphasense,  
<https://zueriluft.ch/makezurich/AAN803.pdf>
3. Gonzalez, A., "Field Calibration of Low-Cost Air Pollution Sensors",  
<https://www.atmos-meas-tech-discuss.net/amt-2019-299/amt-2019-299.pdf>
4. Mijling, B., "Field calibration of electrochemical NO<sub>2</sub> sensors in a citizen science context", <https://www.atmos-meas-tech.net/11/1297/2018/amt-11-1297-2018.pdf>
5. "Kalibratie van NO<sub>2</sub>-sensoren", Samen meten aan luchtkwaliteit,  
<https://www.samenmetenaanluchtkwaliteit.nl/documenten/kalibratie-van-no2-sensoren>
6. Wesseling, J., "Meten van NO<sub>2</sub> met goedkope sensoren",  
<https://www.samenmetenaanluchtkwaliteit.nl/sites/default/files/2017-09/RIVM%20notitie%20meten%20van%20NO2%20met%20goedkope%20sensoren.pdf>
7. Hagan, D. H., "Calibration and assessment of electrochemical air quality sensors by co-location with reference-grade instruments",  
<https://dspace.mit.edu/bitstream/handle/1721.1/115417/amt-11-315-2018.pdf?sequence=1&isAllowed=y>
8. "Airtime Calculator", The Things Network,  
<https://www.thethingsnetwork.org/airtime-calculator>
9. Bäumker, E., "Minimizing power consumption of LoRa® and LoRaWAN for low-power wireless sensor nodes", <https://iopscience.iop.org/article/10.1088/1742-6596/1407/1/012092/pdf>
10. N. Labordus, "Design Sense2Grow GEN node," Technolution, Gouda, 2020

Bijlage D

# Testrapport

# Verbeterde firmware GEN-node



## Testrapport

*Mei 2020, Gouda*

**Opdrachtgever**

Olaf Peters

**Auteur**

Jeremy Hut



DE HAAGSE  
HOGESCHOOL



# Inhoudsopgave

<b>1. Inleiding</b>	<b>2</b>
<b>2. Testaanpak</b>	<b>3</b>
<b>3. Testplan</b>	<b>4</b>
3.1 Iteratie 1: Analoog	4
3.2 Iteratie 2: Luchtkwaliteit	8
3.3 Iteratie 3: Energiezuinigheid	10
<b>4. Uitgevoerde test</b>	<b>12</b>
4.1 Iteratie 1: Analoog	12
4.2 Iteratie 2: Luchtkwaliteit	16
4.3 Iteratie 3: Energiezuinigheid	18

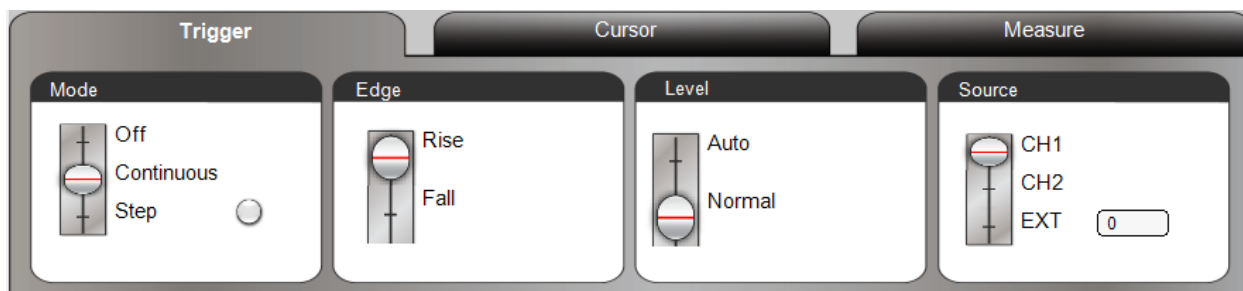
# 1. Inleiding

Het testrapport bevat de testen die uitgevoerd zijn op de GEN-node met de verbeterde firmware. Dit is in 3 iteraties gedaan. Bij elke iteratie horen enkele requirements. Uiteindelijk komt elk functionele requirement aan bod. Er wordt beschreven hoe de test uitgevoerd moet worden en er wordt beschreven wat het resultaat is van de test.

## 2. Testaanpak

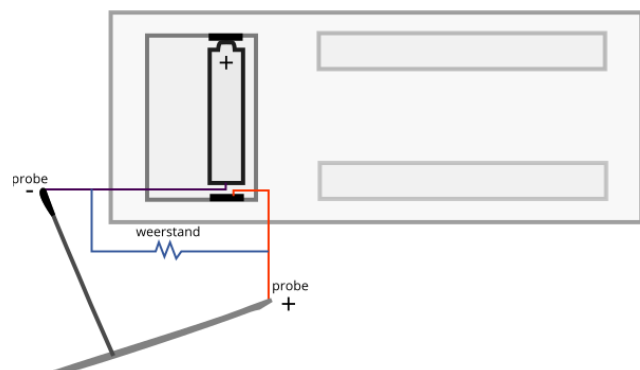
Bij de testaanpak is opgenomen hoe het testen in hoofdstuk 4 is uitgevoerd. Bij het meten van spanningen van de sensor wordt gebruik gemaakt van een oscilloscoop. Hiervoor moet de sensor aangesloten worden op de GEN-node volgens het ontwerprapport. Voor een generieke analoge sensor betekent dit op ADC2. Sluit de probe van de oscilloscoop aan door in dit geval de positieve pool te koppelen aan ADC2 en de negatieve pool op GND.

De oscilloscoop zal in de meeste testen een spanning moeten meten gedurende slechts enkele milliseconden tot tientallen milliseconden. Maak hiervoor gebruik van een trigger bij een opgaande flank (afbeelding 1). Veel sensoren worden (alleen) tijdens de meting gevoed op de poort 3V3\_OUT. Door een tweede probe aan te sluiten op deze poort. Dit is een ideaal triggerpunt. Stel de oscilloscoop zodanig in dat verticale as tijdstappen bevat van rond de 10ms.



» Afbeelding 1, trigger instellen op de PropScope USB Oscilloscoop

Het gebruikmaken van triggerpunten is ook handig bij het kunnen meten van stroom in korte perioden. Plaats de eerste probe met de plus en de minpool binnen de stroomkring. Een handige plaats is tussen de minpool van de batterij en de minpool van de batterijhouder. Scherm de verbinding hiertussen af met isolatietape. Parallel tussen deze twee polen van de probe komt een weerstand. Met de wet van Ohm ( $I=U/R$ ) kan de meting op het scherm omgerekend worden naar stroom.



» Afbeelding 2, schematische tekening stroommeten op oscilloscoop

## 3. Testplan

### 3.1 Iteratie 1: Analooog

**Beschrijving:** De firmware heeft een implementatie voor het aansturen en het uitlezen van de externe ADC.

**R02**

**Precondities:** UART wordt uitgelezen, DIP-switch staat op 0000

**Acties:**

- Sluit 1400 mV aan op de poort ADC0P en de negatieve pool op ADC0N
- Zet de GEN-node aan

**Verwacht resultaat:**

Console geeft: `New sample from sensor: 1400mV`

**Daadwerkelijk resultaat:**

**OK**

**Beschrijving:** De firmware geeft de keuze voor het gebruik van analoge sensoren, zodat er ondersteuning is voor een sensoren met een uitgangssignaal van 4-20 mA, 0-2,5 V, 0-5 V en 0-10 V.

**R03**

**Precondities:** UART wordt uitgelezen, GEN-node staat uit, DIP-switch staat op 0000

**Acties:**

- Sluit een 4-20 mA sensor aan
- Start de node
- Sluit een sensor aan van 0-2,5 V, 0-5 V en daarna 0-10 V en herstart tussendoor de node.

**Verwacht resultaat:**

Een sample van de sensor in mV die overeenkomt met de uitgangsspanning van de sensor

**Daadwerkelijk resultaat:**

OK

**Beschrijving:** De firmware geeft de keuze voor het toepassen van de benodigde gain, zodat er ondersteuning is voor analoge sensoren met een zwak uitgangssignaal.

**R04**

**Precondities:** UART wordt uitgelezen, GEN-node staat uit

**Acties:**

- Sluit de sensor SQ-500 aan (0 – 40 mV sensor)
- Zet de DIP-switch op 0101 (analoog – versterking 32x)
- Zet de node aan

**Verwacht resultaat:**

Console geeft: **Gain is set to 32x**

Samplewaarde in de console komt overeen met gemeten waarde

**Daadwerkelijk resultaat:**

**OK**

**Beschrijving:** Uit de ondersteunde ingangssignalen van sensoren kan worden gekozen zonder dat hiervoor gewisseld hoeft te worden van firmware.

**R06**

**Precondities:** UART wordt uitgelezen, GEN-node staat uit

**Acties:**

- Zet de DIP-switch op 0000
- Lees de meting en eventuele waarschuwingen van het scherm
- Verander de DIP-switch naar 1000
- Reset de node door de magneet langs 'reset' te halen
- Lees de meting en eventuele waarschuwingen van het scherm
- Verander de DIP-switch naar 1001 en herhaal het patroon totdat er tot DIP-switch-intelling 1111 is gemeten.

**Verwacht resultaat:**

Op de console verschijnt een meting van de sensor.

**Daadwerkelijk resultaat:**

**OK**

## 3.2 Iteratie 2: Luchtkwaliteit

**Beschrijving:** De firmware heeft ondersteuning voor geselecteerde driedraadssensoren (NO<sub>2</sub> en SO<sub>2</sub>).

**R05**

**Precondities:** UART wordt uitgelezen, DIP-switch staat op 1000, alternatieve spanningsmeetmethode aangesloten

**Acties:**

- Sluit een NO2-B43F of SO2-B4 aan op de poorten beschreven in het Ontwerprapport
- Zet de GEN-node aan
- Wacht op de 2<sup>e</sup> meting.

**Verwacht resultaat:**

Console geeft het meetresultaat in mV en is kloppend

**Daadwerkelijk resultaat:**

**OK**



**Beschrijving:** De firmware heeft de keuze een kleine uitgangsstroom in te schakelen van 0,5mA waarmee de PT500-sensor gevoed kan worden.

**R07**

**Precondities:** UART wordt uitgelezen, DIP-switch staat op 1001

**Acties:**

- Sluit een RTD of soortgelijke weerstand aan op de poorten ADC0N (L1), ADC0P (L2) en REFIN (L3).
- Zet de GEN-node aan

**Verwacht resultaat:**

Console een meetresultaat in mV dat overeenkomt met de uitgangsspanning van de sensor

**Daadwerkelijk resultaat:**

**OK**

### 3.3 Iteratie 3: Energiezuinigheid

**Beschrijving:** De firmware heeft driverondersteuning voor de ingangen Analog In, SPI en I2C en de uitgangen Analog Out en Digital Out.

**R01**

**Precondities:** UART wordt uitgelezen

**Acties:**

- Sluit een ondersteunde I2C-sensor aan en stel DIP-switch in.
- Zet de GEN-node aan
- Bekijk de meting
- Sluit een PT500 of soortgelijke sensor aan en zet de DIP-switch op 1001
- Reset de GEN-node met de magneet
- Bekijk de meting

**Verwacht resultaat:**

Console geeft aan dat het uitlezen van de I2C-sensor is voltooid door een meetwaarde te tonen (*I2C en Digital Out*). Het uitlezen van een PT500 is gelukt door het tonen van een acceptabele meetwaarde: tussen 0,1 en 2,4 V (*SPI, Analog In en Analog Out*).

**Daadwerkelijk resultaat:**

**OK**

**Beschrijving:** Het energieverbruik is lager dan het huidig verbruik wanneer de GEN-node wordt toegepast voor het uitlezen van analoge sensoren met de externe ADC.

**R09**

**Precondities:** Sluit binnen de stroomkring een oscilloscoop aan. Configureer de opstelling:

- Probe 1 binnen de stroomkring aansluiten met kleine weerstand (ordegrootte 200 mV)
- Probe 2 aansluiten op 3v3\_switched (ordegrootte 2V)
- Tijdvakken op 10 ms
- Trigger op omhooggaande flank van probe 2

**Acties:**

- Plaats de nieuwe firmware op de GEN-node
- Zet de GEN-node aan op DIP-stand 0000 (analoge sensor, gain 1)
- Wacht totdat de ADC voor de tweede keer heeft gemeten
- Bekijk de meting van de oscilloscoop
- Reken de spanningen van probe 1 om naar stroom (wet van Ohm)

**Verwacht resultaat:**

De oppervlakte onder de grafiek is kleiner dan 0,00016 mAh.

**Daadwerkelijk resultaat:**

**OK**

## 4. Uitgevoerde test

### 4.1 Iteratie 1: Analooog

**Beschrijving:** De firmware heeft een implementatie voor het aansturen en het uitlezen van de externe ADC.

**R02**

**Precondities:** UART wordt uitgelezen, DIP-switch staat op 0000

**Acties:**

- Sluit 1400 mV aan op de poort ADC0P en de negatieve pool op ADC0N
- Zet de GEN-node aan

**Verwacht resultaat:**

Console geeft: `New sample from sensor: 1400mV`

**Daadwerkelijk resultaat:**

Console geeft 1399 mV

**OK**   **Ja**

**Beschrijving:** De firmware geeft de keuze voor het gebruik van analoge sensoren, zodat er ondersteuning is voor een sensoren met een uitgangssignaal van 4-20 mA, 0-2,5 V, 0-5 V en 0-10 V.

**R03**

**Precondities:** UART wordt uitgelezen, GEN-node staat uit, DIP-switch staat op 0000

**Acties:**

- Sluit een 4-20 mA sensor aan
- Start de node
- Sluit een sensor aan van 0-2,5 V, 0-5 V en daarna 0-10 V en herstart tussendoor de node.

**Verwacht resultaat:**

Een sample van de sensor in mV die overeenkomt met de uitgangsspanning van de sensor

**Daadwerkelijk resultaat:**

De analoge sensoren worden uitgelezen, het resultaat in mV is vergelijkbaar met de alternatieve meting op een spanningsmeter.

**OK**   **Ja**

**Beschrijving:** De firmware geeft de keuze voor het toepassen van de benodigde gain, zodat er ondersteuning is voor analoge sensoren met een zwak uitgangssignaal.

**R04**

**Precondities:** UART wordt uitgelezen, GEN-node staat uit

**Acties:**

- Sluit de sensor SQ-500 aan (0 – 40 mV sensor)
- Zet de DIP-switch op 0101 (analoog – versterking 32x)
- Zet de node aan

**Verwacht resultaat:**

Console geeft: **Gain is set to 32x**

Samplewaarde in de console komt overeen met gemeten waarde

**Daadwerkelijk resultaat:**

Console geeft Gain is set to 32x.

Samplewaarde is vergelijkbaar met gemeten waarde.

**OK** **Ja**

Opmerking: De firmware handelt dit goed af, maar de ADC op GEN-node met typenummer 0002 voert de versterking te licht uit.

**Beschrijving:** Uit de ondersteunde ingangssignalen van sensoren kan worden gekozen zonder dat hiervoor gewisseld hoeft te worden van firmware.

**R06**

**Precondities:** UART wordt uitgelezen, GEN-node staat uit

**Acties:**

- Zet de DIP-switch op 0000
- Lees de meting en eventuele waarschuwingen van het scherm
- Verander de DIP-switch naar 1000
- Reset de node door de magneet langs 'reset' te halen
- Lees de meting en eventuele waarschuwingen van het scherm
- Verander de DIP-switch naar 1001 en herhaal het patroon totdat er tot DIP-switch-intelling 1111 is gemeten.

**Verwacht resultaat:**

Op de console verschijnt een meting van de sensor.

**Daadwerkelijk resultaat:**

Zowel analoge sensoren als de opgegeven specifieke sensoren zijn uit te lezen op de console.

**OK**   **Ja**

Opmerking: Hierbij vervalt wel de ERPC-functionaliteit en de informatieve logging op de UART wegens gebrek aan Flash-geheugen.

## 4.2 Iteratie 2: Luchtkwaliteit

**Beschrijving:** De firmware heeft ondersteuning voor geselecteerde driedraadssensoren (NO<sub>2</sub> en SO<sub>2</sub>).

**R05**

**Precondities:** UART wordt uitgelezen, DIP-switch staat op 1000, alternatieve spanningsmeetmethode aangesloten

**Acties:**

- Sluit een NO2-B43F of SO2-B4 aan op de poorten beschreven in het Ontwerprapport
- Zet de GEN-node aan
- Wacht op de 2<sup>e</sup> meting.

**Verwacht resultaat:**

Console geeft het meetresultaat in mV en is kloppend

**Daadwerkelijk resultaat:**

De gemeten waarde in de console is vergelijkbaar met de gemeten waarde op de multimeter

**OK**   **Ja**



**Beschrijving:** De firmware heeft de keuze een kleine uitgangsstroom in te schakelen van 0,5mA waarmee de PT500-sensor gevoed kan worden.

**R07**

**Precondities:** UART wordt uitgelezen, DIP-switch staat op 1001

**Acties:**

- Sluit een RTD of soortgelijke weerstand aan op de poorten ADC0N (L1), ADC0P (L2) en REFIN (L3).
- Zet de GEN-node aan

**Verwacht resultaat:**

Console een meetresultaat in mV dat overeenkomt met de uitgangsspanning van de sensor

**Daadwerkelijk resultaat:**

Console geeft een meetresultaat in mV dat vergelijkbaar is met de gemeten waarde op de multimeter.

**OK**   **Ja**

## 4.3 Iteratie 3: Energiezuinigheid

**Beschrijving:** De firmware heeft driverondersteuning voor de ingangen Analog In, SPI en I2C en de uitgangen Analog Out en Digital Out.

**R01**

**Precondities:** UART wordt uitgelezen

**Acties:**

- Sluit een ondersteunde I2C-sensor aan en stel DIP-switch in.
- Zet de GEN-node aan
- Bekijk de meting
- Sluit een PT500 of soortgelijke sensor aan en zet de DIP-switch op 1001
- Reset de GEN-node met de magneet
- Bekijk de meting

**Verwacht resultaat:**

Console geeft aan dat het uitlezen van de I2C-sensor is voltooid door een meetwaarde te tonen (*I2C en Digital Out*). Het uitlezen van een PT500 is gelukt door het tonen van een acceptabele meetwaarde: tussen 0,1 en 2,4 V (*SPI, Analog In en Analog Out*).

**Daadwerkelijk resultaat:**

De sensoren geven acceptabele meetwaardes waarmee de driverondersteuning is aangetoond.

**OK**   **Ja**

**Beschrijving:** Het energieverbruik is lager dan het huidig verbruik wanneer de GEN-node wordt toegepast voor het uitlezen van analoge sensoren met de externe ADC.

**R09**

**Precondities:** Sluit binnen de stroomkring een oscilloscoop aan. Configureer de opstelling:

- Probe 1 binnen de stroomkring aansluiten met kleine weerstand (scope instellen op ordegrootte 200 mV)
- Probe 2 aansluiten op 3v3\_switched (ordegrootte 2V)
- Tijdvakken op 10 ms
- Trigger op omhooggaande flank van probe 2

**Acties:**

- Plaats de nieuwe firmware op de GEN-node
- Zet de GEN-node aan op DIP-stand 0000 (analoge sensor, gain 1)
- Wacht totdat de ADC voor de tweede keer heeft gemeten
- Bekijk de meting van de oscilloscoop
- Reken de spanningen van probe 1 om naar stroom (wet van Ohm)

**Verwacht resultaat:**

De oppervlakte onder de grafiek is kleiner dan 0,00016 mAh.

**Daadwerkelijk resultaat:**

De oppervlakte is 0,00005 mAh. Dit is gemeten met een weerstand van 3 ohm.

**OK**   **Ja**

Bijlage E

# Adviesrapport

# Verbeterde firmware GEN-node



## Adviesrapport

*Mei 2020, Gouda*

**Opdrachtgever**

Olaf Peters

**Auteur**

Jeremy Hut



DE HAAGSE  
HOGESCHOOL

# Inhoudsopgave

<b>1. Inleiding</b>	<b>2</b>
<b>2. Conclusies</b>	<b>3</b>
<b>3. Aanbevelingen</b>	<b>4</b>
<b>4. Overdracht</b>	<b>6</b>

# 1. Inleiding

In dit adviesrapport zijn de conclusies en aanbevelingen opgesteld. Tot slot volgt er de benodigde informatie voor de overdracht naar Technolution. De adviezen hebben betrekking op het afstudeerproject waarin de firmware van de GEN-node verbeteringen krijgt op het gebied van ondersteuning van analoge sensoren, het kunnen maken van een sensorkeuze, ondersteuning van NO<sub>2</sub>- en SO<sub>2</sub>-sensoren van Alphasense en het energieverbruik.

## 2. Conclusies

De hoofdvraag van dit onderzoek was “Hoe kan de firmware van de GEN-node verbeterd worden?”. De beantwoording van deze probleemstelling is opgesplitst in drie deelvragen.

### **Hoe kunnen analoge sensoren universeel worden ondersteund?**

Door middel van de firmware dat ontworpen en ontwikkeld is in hoofdstuk 6.1 en 6.2 kunnen de gewenste analoge sensoren uitgelezen worden op de GEN-node. Dat is gedaan door aansturing te ontwikkelen voor de externe ADC waarvan de aansturing een groot aantal analoge sensoren ondersteund zoals een 4-20 mA-sensor en de 0-2,5 volt-sensor. Met de DIP-switch is op de GEN-node een keuze te maken in het gewenste lijstje sensoren.

### **Hoe kunnen geselecteerde sensoren ten behoeve van het meten van luchtkwaliteit worden ondersteund?**

Hierbij is een specifieke aansturing voor de geselecteerde Alphasense-sensoren ontworpen en ontwikkeld in hoofdstuk 7.1 en 7.2. Het grootste verschil met de voorgaande deelvraag is dat deze sensoren twee elektroden bevatten die los van elkaar gemeten moeten worden. Dit is nodig voor een juiste omzetting van millivolts naar parts-per-billion.

### **Hoe kan het energieverbruik worden verlaagd?**

Het energieverbruik is verlaagd door de informatieve logging uit te zetten. Voor het uitlezen van een analoge sensor is het energieverbruik extra verlaagd door het hoger zetten van de datarate van de ADC. Het energieverbruik zou verder nog verlaagd kunnen worden wanneer metingen opgespaard worden voordat ze verzonden worden.

De functionele requirements R01 tot en met R08 zijn getest en voldaan. R09 is een could-have en is niet aan bod gekomen. De niet-functionele requirement R09 is getest en met R10, R11 en R12 is rekening gehouden tijdens het proces.

Het doel “De afstudeeropdracht leidt tot verbeterde firmware voor een Proof-of-Concept-model GEN-node die ondersteuning heeft voor analoge sensoren, plus geselecteerde sensoren voor het meten van luchtkwaliteit waarbij het energieverbruik verlaagd is.” is hiermee gehaald. Het verlagen van het energieverbruik is nog niet bereikt tijdens het schrijven van het voorlopige afstudeerdossier.



## 3. Aanbevelingen

### **Grotere LoRa-module**

Om in de toekomst nieuwe sensoren te kunnen toevoegen aan de multi\_sensor-applicatie, of om de eRPC-functionaliteit bij deze applicatie weer te kunnen gebruiken raad ik aan over te stappen naar een andere LoRa-module met een groter flash-geheugen. De fabrikant van de huidige LoRa-module raadt de WiMod iM881A-XL aan. Deze wordt verwacht in het tweede of derde kwartaal van 2020. Deze heeft 192 kB flash. Onbekend is nog hoe groot de statische bibliotheek is.

### **Metingen opsparen**

Metingen opsparen zorgt voor een kortere totale airtime en dus een lager energieverbruik. Door de waarde van \_nrOfSamplesBeforeSending te veranderen kan dit worden ingeschakeld. Omdat elke situatie anders is, staat deze waarde standaard op 1. Afhankelijk van de requirements van de klant zou deze waarde vergroot kunnen worden.

### **Logging uit voor productieomgeving**

Wanneer debugging niet nodig is, raad ik aan om in ieder geval de informatieve logging uit te zetten. Dit bespaart energie en bespaart flash-geheugen. Voor debugging op locatie is het handig om alvast twee versies van de firmware-applicatie te hebben: één met logging en één zonder logging. Voor de applicatie multi\_sensor is dit lastiger, aangezien het inschakelen van logging niet past binnen de 128 kB flash-geheugen. Maak daarom gebruik van twee debugversies, met elk een gedeelte van de ondersteunde sensoren.

### **Backoffice omrekeningen**

Metingen van analoge sensoren worden nu verstuurd naar de backoffice in millivolts. Voor een grotere gebruiksvriendelijkheid kan hier in de backoffice geprogrammeerd worden dat er voor een GEN-node een omrekenformule kan worden opgegeven. De vertaalt de waardes in millivolt naar een nuttige eenheid voor de aangesloten sensor.

### **Kalibratie Alphasense-sensoren**

De fabriekskalibratiewaarden bleken bij testen onvoldoende. Regelmatig herkalibreren is nodig om nuttige informatie uit deze sensoren te achterhalen. De nieuwe kalibratiewaardes kunnen vervolgens geplaatst worden in de omrekenformule die bij een specifieke GEN-node wordt opgeslagen in de backoffice.

**Eventueel vervolgproject**

Een interessant onderwerp voor een vervolgproject is de GEN-node nog generieker maken op het gebied van ondersteuning van sensoren. Wat nu gedaan is met analoge sensoren zou dan uitgebreid kunnen worden naar SPI-sensoren en I<sup>2</sup>C-sensoren.

## 4. Overdracht

De nieuwe software is geüpdatet in SVN. De broncode is hier te vinden in “trunk/software”. De nieuwe applicaties analog\_sensor en multi\_sensor worden gecompileerd onder de nieuwe hardwareversie HSI\_A4. Het verschil met HSI\_A2 is feit dat de poort 3V3\_switched veranderd is van normally closed naar normally open.

Het afstudeerverslag en de overige documentatie zal ook worden gearhiveerd in SVN. Deze verschijnt op de locatie "trunk/documents/design/s2g\_gen\_node\_base/2020-02 Jeremy Hut". Hoe de sensoren aangesloten worden met huidige aansluitingen is te vinden in het ontwerprapport.