Labo Embedded Systems II

Team TurnTable

Daan Delabie - Pieter Vanherck - Tuur Vandecauter - Jona Cappelle – Niels Bauwens

2019 - 2020

# Conceptbeschrijving

Op het terras bij het studentenrestaurant van Technologiecampus Gent zijn enkele tafels voorzien om buiten te zitten, eten, werken, … . Studenten blijven echter niet lang zitten. Met project ‘Turn Table’ wordt ervoor gezorgt dat mensen langer blijven zitten aan de voorziene tafels. Hierdoor wordt een leuke campus sfeer gecreëerd. Om dit te doen wordt van de tafel een interactief spel gemaakt die gebaseerd is op het spel ‘Simon’. Dit is een geheugenspel waarbij een bepaalde sequentie van vier verschillende symbolen wordt weergegeven. De speler moet de combinatie van die symbolen herhalen. Indien de speler hierin slaagt wordt een extra symbool toegevoegd aan de sequentie. Dit gaat verder tot de speler een fout maakt. Zijn score is gelijk aan het maximum aantal symbolen die de speler onthouden heeft.

De tafels zijn voorzien van een draaiend platform in het midden. Hiervan wordt gebruik gemaakt om het spel een nieuwe dimensie te geven. Het volledig systeem wordt ingewerkt in de schijf. Indien de schijf beweegt wordt met behulp van een accelerometer de aanwezigheid van een speler vastgelegd. Zo kan het systeem uit sleepmode gehaald worden. De positie waarin de schijf gedraaid staat bepaalt het symbool. Om de sequentie van symbolen te vormen in het spel moet aan de schijf gedraaid worden tot deze op een bepaalde posities staat, dit voor ieder symbool. Er wordt een drukknop voorzien waarbij de speler de symbool positie van de schijf kan bevestigen. De positie van de schijf wordt bepaald aan de hand van een kompas sensor.

De speler is in staat om zijn score af te lezen via 7-segment displays. Indien de score binnen de top 3 valt wordt deze doorgestuurd via LoRa zodat de top 3 altijd online te raadplegen valt. Om een identiteit te geven aan de spelers uit de top 3 wordt vooraleer het spel te starten de studentenkaart van de speler gescand via RFID. Op die manier kunnen ook enkel studenten afkomstig van de campus dit spel spelen.

Het volledige embedded system werkt autonoom met behulp van een batterij en een zonnepaneel.

Bij dit ontwerp wordt rekening gehouden met het retrofitten van de tafel, die bovendien onderhevig zijn aan de belgische weersomstandigheden, waarbij gestreefd wordt naar een esthetisch mooi design.

# Specificaties

Functionele vereisten

* Autonome werking voor minimum 5 jaar
* Data moet verzonden worden na ieder gespeeld spel waarbij de behaalde score binnen de top 3 valt
* Data moet altijd online beschikbaar zijn met een performantie van 90%
* Communicatie link moet een performantie hebben van 90%
* Het spel moet 97% van de tijd goed werken
* Het spel moet gedurende een uur kunnen gespeeld worden zonder stoppen
* De positie van de schijf moet  geregistreerd worden met een afwijking tussen -35° tot  35° gemeten tussen de symbool positie en de pijl.
* De speler moet een score kunnen zien
* De speler moet een voorbeeld sequentie kunnen zien van het “simon” spel
* De tafel moet eender waar buiten op de technologiecampus gent functioneren
* Spel moet op duidelijke manier gespeeld kunnen worden zonder veel extra uitleg

Andere technische vereisten

* Retrofitten van een bestaande tafel die buiten staat, met in het midden een draaiende schijf
* Waterdicht IP46  (bestendig tegen stortbuien)
* Bestand tegen rukwinden van 100 km/u
* Bestand tegen buitentemperaturen in alle seizoenen (-15 tot 45 °C)
* Gewicht van maximum 5 kg
* Moet werken in omstandigheden waarbij de zonnestraling volgende waarden heeft in België: Gemiddeld 530 Wh/m² in december en 5070 Wh/m² in juni, gemeten op een horizontaal oppervlak

Niet-technische vereisten

* Esthetisch mooi verwerkt in de voorziene tafel
* anti diefstal
* Wegneembaar/afneembaar
* Kostprijs van maximum € 100
* Deadline werkend prototype: 7 mei 2020
* Er mag geen schade aangebracht worden aan de tafel (niet boren, zagen, schijven, frezen, …)
* De speler moet het spel kunnen spelen met een goede ergonomische houding, al zittend op de bank bij de tafel

# Sensoren

Voor de sensoren werden een accelerometer en kompas geselecteerd als meest geschikt voor deze toepassing. Er bestaan IMU’s ( Inertial measurement units) met als doel bewegingen te meten. Voor de kleine package en de bijna identieke prijs van het kopen van een accelerometer en kompas apart, werd hier voor de IMU gekozen. Het voordeel van de IMU is dat deze ook een gyroscoop aan boord heeft (waar in de toekomst eventueel ook nog gebruik van zou kunnen gemaakt worden). De IMU die geselecteerd werd was de ICM20948 vanwege zijn zeer laag vermogen verbruik.

Deze chip werkt op 1.8 V en verbuikt onder andere daardoor slechts 2.5 mW. Hierdoor zal er in het hardware design ook level shifting en een spanningsregelaar voorzien worden. De chip beschikt over een accelerometer met programmeerbare FSR van ±2g , ±4g, ±8g en ±16g en een kompas met een groot bereik van ±4900 µT. Deze waarden worden omgezet door on-chip 16-bit ADCs. De chip beschikt ook over een digitale temperatuursensor om temperatuurcorrecties door te voeren. Deze kan ook uitgelezen worden en mogelijks gebruikt worden om bijvoorbeeld de buitentemperatuur te monitoren en mee door te sturen.

## Software

Om de samenvoeging van de verschillende onderdelen in dit project vlot te laten verlopen werden een aantal functies geschreven om sensorwaarden uit te lezen. Deze werden ondergebracht in een library. Door deze library te includen, de juiste pinnen aan te passen en de voorgestelde functies te gebruiken kan op een minimum van tijd geïnterfaced worden met de sensor.

Functies die geschreven werden zijn:

* ICM\_20948\_Init();

Een functie voor het initialiseren van de ICM20948 sensor, dit bevat zowel de gyroscoop, accelerometer als het kompas. Hier wordt ook instellingen van de I2C interface correct gezet en de communicatie wordt gestart. Er wordt nagegaan of de sensor wel degelijk verbonden is door het “Who am I” register uit te lezen. Ook sample rates worden hier ingesteld. Het kompas (dat een aparte chip is in de ICM20948) wordt door middel van een I2C bypass multiplexer rechtstreeks verbonden met de I2C bus.

* ICM\_20948\_accelDataRead(data.ICM\_20948\_accel);

Een functie om de accelerometer waarden uit te lezen in drie dimensies (x, y en z). Voor het uitlezen van alle waarden wordt er gebruik gemaakt van een “struct” (hier genaamd: data), een soort structure die alle uitgelezen waarden bijhoudt.

* ICM\_20948\_magDataRead(data.ICM\_20948\_magn);

Een functie die de kompas waarden uitleest en omzet naar µT, ook in drie dimensies. Dit gebeurt aan 100 Hz.

* ICM\_20948\_magn\_to\_angle(data.ICM\_20948\_magn, data.angle);

Een functie die de “raw” kompas waarden omzet naar een hoek in graden. Deze hoek kan variëren tussen -180° en +180° en kan gebruikt worden om de positie van de schijf te bepalen.

* ICM\_20948\_calibrate\_mag(data.offset, data.scale);

Een functie die het kompas kalibreert. Deze waarden worden opgeslagen en worden na kalibratie ook mee in rekening gebracht in de functie: “ICM\_20948\_magDataRead” en ”ICM\_20948\_magn\_to\_angle”. Er is dus een volledige integratie in de geschreven library en er zijn achteraf geen correcties van offsets meer nodig.

* ICM\_20948\_wakeOnMotionITEnable(true, 50, 50);

Een functie die alle sensoren uit zet en enkel de accelerometer aanzet in low power mode. Het tweede argument bepaald de sample rate en het derde argument bepaald de threshold in mG’s. De sample rate wordt automatisch omgezet naar een waarde die in het register van de sample rate divider past. (bijvoorbeeld 50 Hz wordt in de praktijk 48.9 Hz) Wanneer de accelerometer een verandering meet groter dan deze threshold zal deze een interrupt genereren en het systeem wakker maken.

De volledige source-code is terug te vinden op GitHub. (Zie referenties)

## Kompas

Er werd gekozen voor een kompas om de oriëntatie van de draaiende schijf te bepalen. Aangezien het aards magnetisch veld zo goed als constant is, kunnen we dit gebruiken als ‘ideale’ referentie voor de positionering. Het kompas dat hiervoor gekozen werd is de ak09916 (geïntegreerd in de ICM20948). Dit kompas kan het aards magnetisch veld meten aan 100 Hz, wat zeker snel genoeg is om het spel te kunnen spelen.

In figuur 1 is een meting te vinden die de correcte werking van het kompas illustreert. In deze meting werd het kompas 360 graden rond zijn as gedraaid. Dit werd gedaan door deze handmatig vast te nemen, door de onregelmatigheden in de draaibeweging is de grafiek niet perfect. Wat we hiermee willen aantonen is dat de nauwkeurigheid wel zeker voldoende is om vier quadranten te onderscheiden van elkaar.

Afbeelding met kaart, man

Automatisch gegenereerde beschrijving

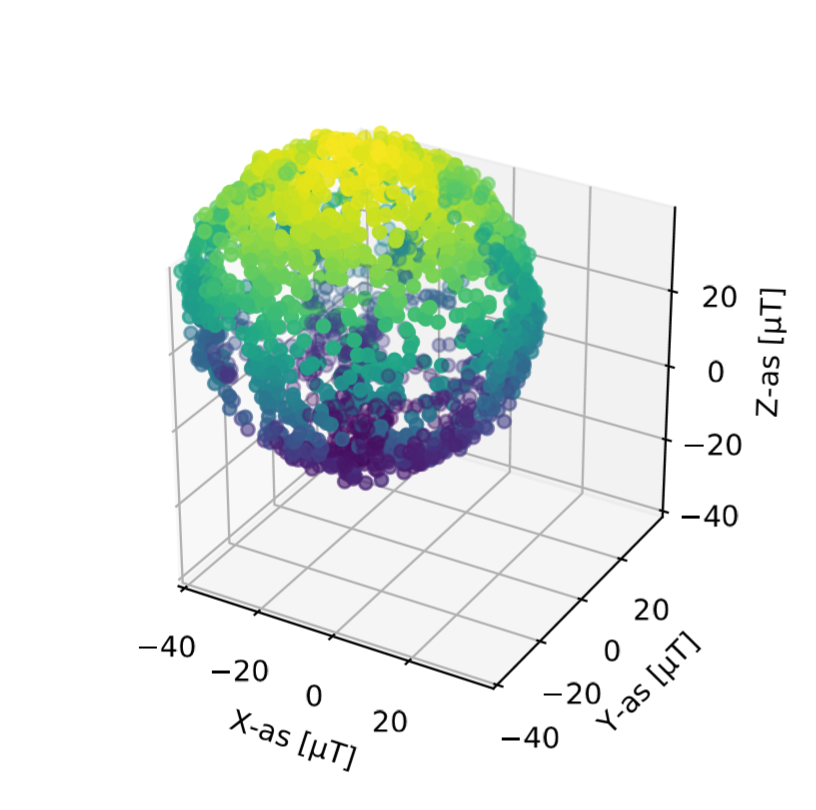
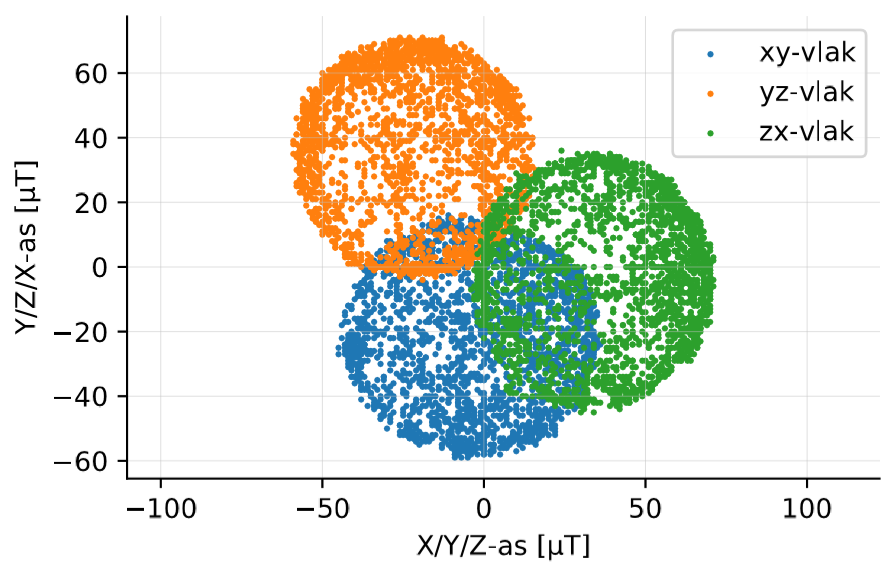
Figuur 1: Meting kompas 360° rotatie

## Kalibratie

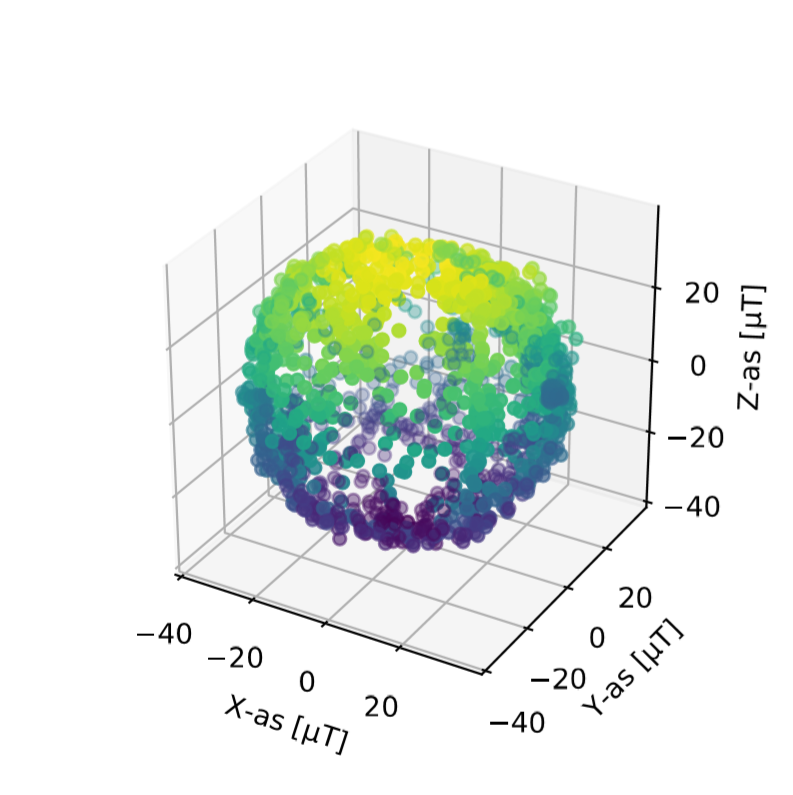
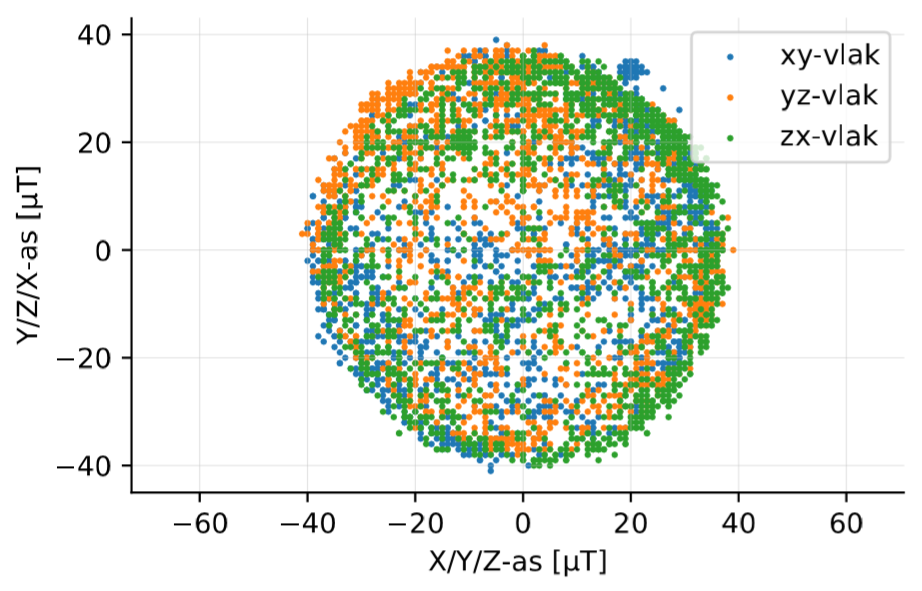
Om nauwkeurige metingen te bekomen is kalibratie van sensoren zeker een meerwaarde. Vooral bij het kompas is dit zeker aan te raden daar deze sensor veel last heeft van fouten. Zo bestaan er twee types vervormingen van de data: harde distorties en zachte distorties.

Harde distorties worden veroorzaakt door objecten die zelf een magnetisch veld creëren, bijvoorbeeld een magneet in een luidspreker. Dit type distorties blijft steeds op eenzelfde plaats ten opzichte van het referentiesysteem van het kompas. Hierdoor krijgt men dus een constante offset. Corrigeren voor harde distorties gebeurt door de offset in x, y en z zichting te bepalen en deze in rekening te brengen bij de metingen. De offset van de verschillende assen is goed te zien op ﬁguur 2. Figuur 4 geeft de sfeer weer na kalibratie. De offsets zijn ook in 3D weergegeven in ﬁguur 3 (voor kalibratie) en ﬁguur 5 (na kalibratie).

Zachte distorties zullen het magnetisch veld op een bepaalde manier vervormen. Bijvoorbeeld zal de x-component meer verzwakt worden dan de y- en de z-component. Dit gebeurt typisch door een materiaal dat een magnetisch veld vervormd maar er zelf geen maakt. Een voorbeeld hiervan is ijzer. Hiervoor corrigeren is moeilijker, is meer rekenintensief en vereist een 3x3 transformatie matrix. In software werd dit geïmplementeerd door gedurende enkele seconden (vb 15 seconden) metingen te doen terwijl men de sensor in alle mogelijke richtingen draait. Hierna wordt voor elke as (x, y en z) de maximale en minimale waarde bepaald. Deze worden genormaliseerd om overeen te komen met een cirkel. De som en het verschil van deze waarden wordt berekend. De offset wordt bepaald aan de hand van de helft van het verschil tussen minimalen maximale waarde. Voor de schaal correctie wordt eerst het gemiddelde van de verschillen van de drie assen berekend en deze gedeeld door het verschil van elke as.



Figuur 2: Assen kompas voor kalibratie Figuur 3: 3D plot voor kalibratie



Figuur 4: Assen kompas na kalibratie Figuur 5: 3D plot na kalibratie

## Accelerometer

Om detectie van aanwezigheden te doen wordt er gebruik gemaakt van een accelerometer. Dit type sensor meet versnellingen. Deze sensor zal ook altijd aan staan. Voor de activiteitsdetectie zou ook voor een gyroscoop kunnen gekozen worden. Dit werd echter niet gedaan omdat een gyroscoop typisch veel meer (enkele mA) verbruikt dan een accelerometer (enkele tientallen µA). De accelerometer zal een interrupt pin hoog zetten wanneer beweging gedetecteerd wordt. Hierdoor zal de microcontroller wakker gemaakt worden, zullen de LED’s gaan branden en kan het spel beginnen.

## Energiemetingen

Door het niet ontvangen te hebben van de ontworpen PCB is het niet mogelijk geweest energiemetingen van deze sensoren te verzamelen. De ICM20948 werd uitgetest en geprogrammeerd aan de hand van een breakout bordje van SparkFun. Hierop werd een LDO geïntegreerd met een zeer grote Iq (lekstroom). Hierdoor zouden geen representatieve metingen verkregen worden aangezien we op de PCB zouden gebruik maken van een LDO met een Iq van 1 µA.

Enkele berekeningen werden wel gedaan: een voorbeeld

In de datasheet is een verbruik van 36.1 µA terug te vinden bij een sample frequentie van 48.9 Hz.

LDO (TPS7A05): Iq = 1 µA (typ), 3 µA (max)

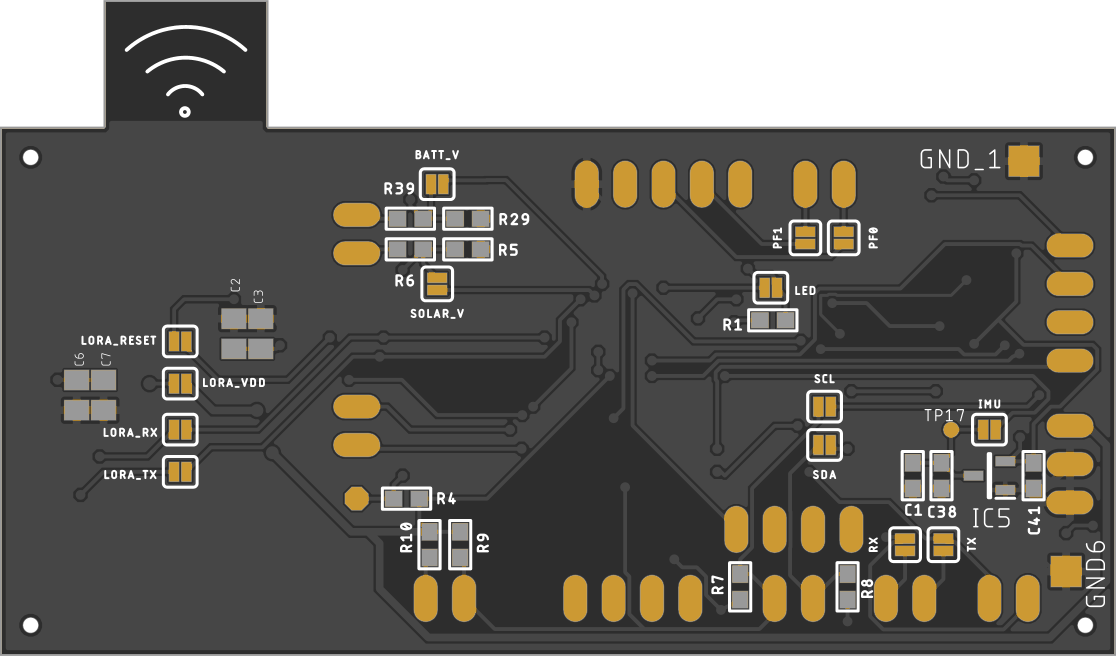
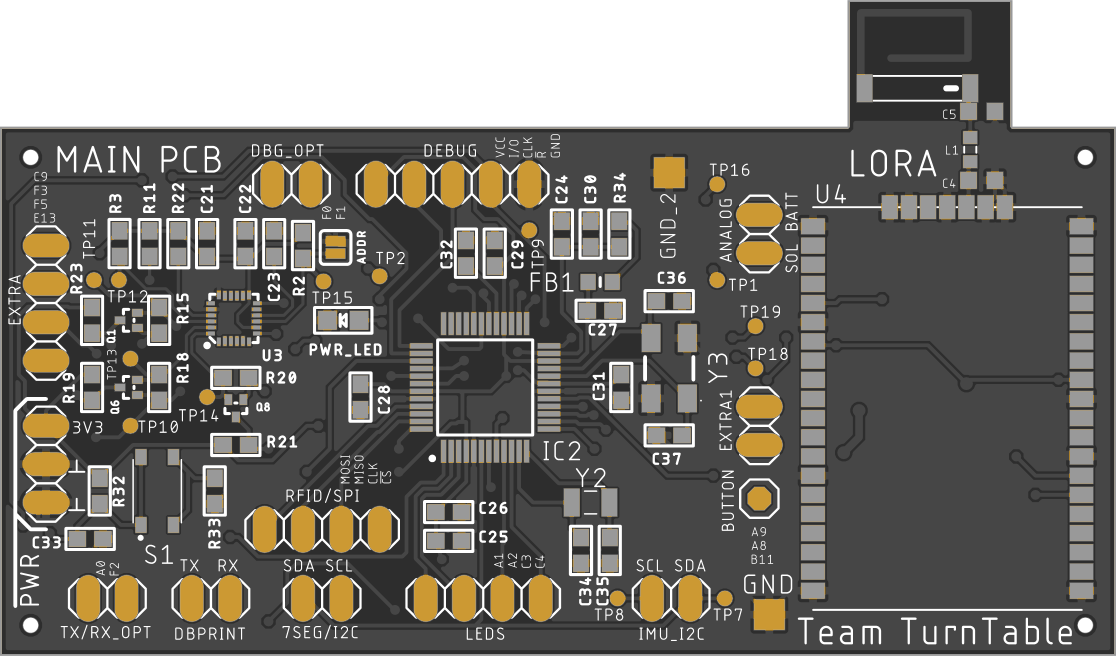
# Design PCB

In figuur 6 en 7 is de ontworpen PCB te zien. Hier werden enkele ontwerpkeuzes gemaakt. Een eerste keuze is het goed scheiden van de draadloze communicatie en het microcontroller en sensoren gedeelte. De antenne van de LoRa module werd naar buiten gebracht om zo een beter bereik te kunnen garanderen. Er werd gekozen voor een antenne op de PCB en er was geen nood aan een antenne die signalen verder kon versturen aangezien we op de campus een LoRa gateway hebben en hier zeer dicht bij zitten.

De microcontroller wordt gevoed op 3.3 V. De 3.3 V komt van het energy harvesting bodje. De sensoren werden ook geïntegreerd op deze PCB. De IMU die gebruikt wordt, werkt op 1.8 V en wordt voorzien door een LDO. Ook de communicatie met de IMU werkt op deze lagere spanning, hiervoor werd ook de gepaste level shifting voorzien. De LoRa communicatie werkt op 3.3 V en aangezien deze veel stroom kan trekken (typisch ±40mA) kon dit niet via een pin uit de microcontroller gevoed worden. Deze module werd rechtstreeks aangesloten op de voedingsspanning. Ook werden de traces hiervoor iets breder genomen.

Op deze PCB werd ervoor gekozen om alle pinnen naar buiten te brengen. Zo kon het gebruikt worden als development bordje voor het prototype. Ook werden tal van testpunten voorzien en jumpers om het debuggen te vergemakkelijken.

Het resultaat is een PCB met afmetingen: 73.9 mm x 43.4 mm, wat zeker klein genoeg is om in de roterende schijf te passen.



Figuur 6: Voorkant main PCB Figuur 7: Achterkant main PCB

# Besluit

# Referenties

De C-code voorzien van gedetailleerde Doxygen documentatie samen met de design files en hardware schema’s is terug te vinden op: <https://github.com/jonacappelle/Embedded-Systems-II-Lab>