



Professionele Bachelor Elektronica-ICT

Hand Gesture Control

Onderdeel van de stage
ondersteund door

AP Hogeschool

en uitgevoerd op en begeleid door het bedrijf

ED&A

Robbe Elsermans

Specialisatie IoT

Inhoudstafel

VERSIEBEHEER.....	4
TERMEN EN AFKORTINGEN	5
OPDRACHTGEVER.....	6
SAMENVATTING.....	7
SITUATIE AS-IS.....	8
PROBLEEMSTELLING	8
SITUATIE TO-BE	9
PROJECTDEFINITIE.....	9
<i>Scope</i>	10
<i>Niet in Scope</i>	11
PLANNING	12
FUNCTIONEEL DESIGN	15
TECHNISCH DESIGN	16
HARDWARE ANALYSE.....	16
<i>Blokdiagram</i>	17
<i>Specificaties</i>	18
<i>Onderliggende argumentatie</i>	19
<i>Elektrisch schema</i>	20
SOFTWARE ANALYSE	23
<i>Data in/ Out</i>	23
<i>State-diagram</i>	24
<i>Flowchart</i>	25
BESCHRIJVING VAN DE MOGELIJKE INTERFACES	26
BESCHRIJVING VAN EVENTUELE IMPACT OP DE HUIDIGE INFRASTRUCTUUR	28
DOCUMENTATIE.....	29
BRONVERMELDING	30

Versiebeheer

Nr.	Datum	Verspreiding	Status	Wijziging
0.1	2022-01-28	/	Eerste draft	Aanmaak document + Inleiding + termen en afkortingen + opdrachtgever + samenvatting + situatie as-is + situatie to-be
0.11	2022-02-04	/	Eerste draft	Blueprint van IoT vergelijken en aanvullen in eigen blueprint
0.2	2022-02-07	/	Eerste draft	Opdrachtgever verduidelijken + samenvatting + Situatie as-is + situatie to-be + planning deels
0.3	2022-02-16	/	Bijna presenteerbaar als draft	Functioneel design + technisch design
0.4	2022-02-18	Philip Fliegers, Kelly Casal Mosteiro	presenteerbaar als draft	Alles zelfde font typen, opkuis lege spaties, lay-out wijzigingen.
0.5	2022-02-18	/	draft	De scope aanpassen
0.6	2022-02-21	/	draft	Aanpassen blokdiagram + toevoegen smart objecten + toevoegen commentaar bij technisch design + elektrisch design
0.7	2022-02-25	Philip Fliegers	Bijna af	Toevoeging interface + infrastructuur + documentatie + algemeen nakijken
0.8	2022-02-28	/	Bijna af	algemeen nakijken + bronnen
1.0	2022-03-03	AP-Hogeschool, Philip Fliegers	Finale versie	Algemeen nakijken + feedback Philip + foto nummers aanpassen

Termen en Afkortingen

Term	Omschrijving
ToF	Time of Flight ([1])
GPIO	General Purpose Input Output
API	Application Programming Interface
PCB	Printed Circuit Board
VR	Voltage Regulators
MCU	Microcontroller Unit
BTN	Drukknop
LED	Light Emitting Diode
FoV	Field of View
I ² C	Inter-Integrated Circuit
SPI	Serial Peripheral Interface
UART	Universal Asynchronous Receiver-Transmitte ([2])
USART	Universal Synchronous-Asynchronous Receiver-Transmitter ([2])
LSB	Least-significant bit
MSB	Most-significant bit
3D	3 Dimensional
EN	Enable
INT	Interrupt
EMC	Electromagnetic Compatibility

Opdrachtgever

ED&A is de opdrachtgever van dit project. Ze zijn gespecialiseerd in het maken van Custom-made elektronica voor consumenten- en industriële toepassingen en gelegen in Kalmthout. Sinds 1981 zijn ze hiermee bezig en blijven tot op heden een grote speler binnenin de Custom-made elektronica.

Men ontwikkelt intern zelf de hardware met de daarbij horende embedded software. Daarbij bezitten ze ook over een eigen hypermoderne geautomatiseerde assemblagelijnen waarvoor ze bekroond werden als **Factory of The Future**. Naast het volledig ontwikkelen van hardware, software en Printed Circuit Boards (PCB's) kunnen ze ook de geassembleerde producten op Electromagnetic Compatibility (EMC) gedeeltelijk Pre-Compliance testen. Hiervoor beschikken ze over allerhande faciliteiten, van klimaatkast tot een geavanceerde EMC-ruimte. Met meer dan 40 jaar ervaring, zijn ze van alle markten thuis. ([3])

De heer Flieger Philip zal dit project begeleiden. Met zijn kennis als Hardware Design Engineer is hij de ideale mentor om dit project in goede banen te leiden. Ook kan ik me berusten op de kennis van het gehele bedrijf en al mijn collega's.

Samenvatting

Het doel van dit project is om een apparaat te ontwikkelen die de user interface een extra dimensie geeft via handbewegingen. Hierdoor is de user niet gelimiteerd tot aanrakingen of spraak maar ook beweging van de handen. De handbewegingen gaan we detecteren met één (of meerdere) “goedkope” Time of Flight (ToF)-sensor(en). De data verkregen van de ToF-sensor(en) zal verwerkt worden met een geschikte Microcontroller (MCU).

De handbewegingen gaan voor deze opstelling simpel zijn zodat de scope niet te groot wordt in dit project. Ook zal er rekening gehouden worden met storing-factoren zoals een cover glas en/ of omgevingslicht. Hiermee verhoogt het applicatiegebied aanzienlijk.

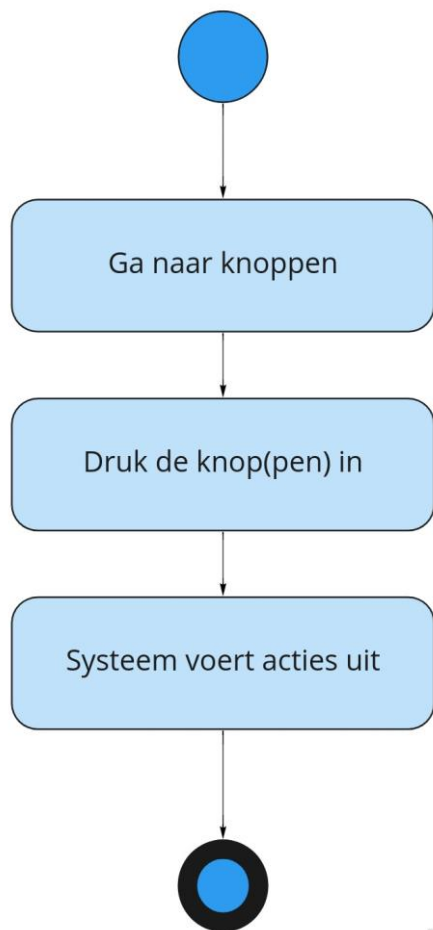
Om de verwerkte data te gebruiken is een interface een must have. Hiermee kan de hand gesture controller gemakkelijk gekoppeld worden aan een apparaat naar keuzen.

Situatie As-Is

Probleemstelling

In huidige situatie is de gebruiker gelimiteerd tot enkelen knoppen of een touchscreen die gebruikt worden om het systeem te bedienen. Omdat er enkel knoppen of een touchscreen aanwezig zal zijn, moet de gebruiker zich telkens verplaatsen naar de desbetreffende interface en deze altijd fysiek aanraken. Dit kan in bepaalde omstandigheden moeilijkheden veroorzaken.

De stappen dat de gebruiker in huidige situatie moet uitvoeren wordt weergegeven in “*Figuur 1 – situatie as-is*”. Een voorbeeld knop gebruikt in elektronica is te zien in “*Figuur 2 – voorbeeld: drukknop*”



Figuur 1 - situatie as-is



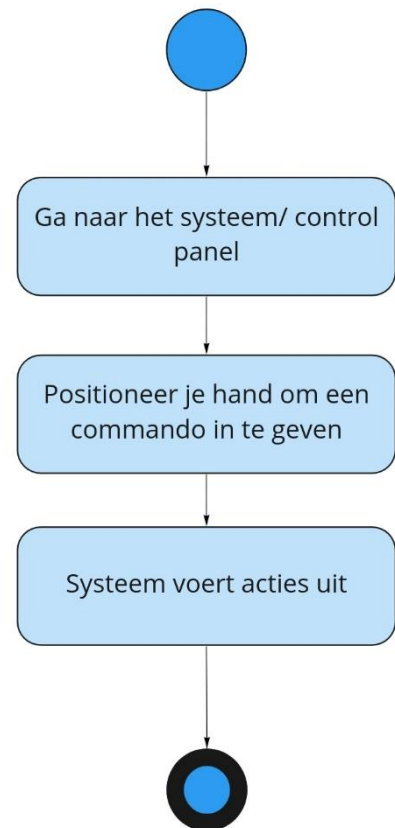
Figuur 2 - voorbeeld: drukknop

Situatie To-Be

Het doel van dit project is om handbeweging herkenbaar te maken voor een apparaat. Het apparaat zal hierdoor weten welke actie er door de user gevraagd wordt en dit uitvoeren (zie "Figuur 3 - situatie to-be").

Dit alles wordt verwezenlijkt met één of meerdere ToF-sensoren en een MCU die deze sensor(en) kan bedienen. De snelheid van het herkennen moet zo hoog mogelijk liggen om de gebruikers ervaring niet te beïnvloeden tijdens het gebruik.

De applicatie moet onderscheid kunnen maken tussen de gegeven hand commando's die verwerkt worden door de MCU. Dit kan finaal in een Application Programming Interface (API) beschreven worden.



Figuur 3 - situatie to-be

Projectdefinitie

- Detecteren van het naderen van een persoon. Hieraan kan een zekere actie/functie gekoppeld worden, zoals bijvoorbeeld het uit stand-by komen van een systeem.
- Detecteren van opdrachten a.d.h.v. beweging met de hand, arm of het lichaam en een apparaat overeenkomstig laten reageren op de uitgebeelde opdracht.

Scope

- **Hoe een ToF-sensor functioneert high level**

Om een actie met een hand te detecteren is het van belang dat we eerst onderzoeken hoe een ToF-sensor in high level opereert. Dit zal de latere uitwerking van het project bevorderen.

- **Onderzoeken welke componenten er geschikt zijn**

In deze opstelling zal er gebruik gemaakt worden van een ToF-sensor en een MCU die de ToF-sensor kan uitlezen.

Belangrijk is dat de gekozen ToF-sensor een tamelijk hoge snelheid voorziet aan metingen en die goedkoop is op de markt. Ook is de maximale afstand dat de ToF-sensor kan meten van belang.

Voor de MCU moeten we kijken naar de kracht van verwerken en of dat het de juiste interface aanboort heeft. De MCU moet ook de gewenste snelheid kunnen hanteren dat de ToF-sensor bied. Doordat het stagebedrijf vele ervaringen heeft met de STM32 MCU's zal er gekeken worden naar deze reeks.

Initieel tests kunnen mogelijk gebeuren op basis van evaluatie kits die deze componenten bevatten.

- **Onderzoek gekozen ToF-sensor low level**

Om de ToF-sensor te kunnen gebruiken gaan we de werking hiervan moeten begrijpen. Met deze kennis kan de communicatie opgezet worden tussen MCU en ToF-sensor. Het kan zijn dat we drivers (API's) ter beschikking krijgen om met de sensor te communiceren.

- **Herken benadering en simpele handbewegingen met 1 sensor**

Na de onderzoeksfase kunnen we starten met het uitwerken van software die een persoon zal detecteren wanneer deze nader komt bij de opstelling (die een actie kan bevatten).

- a. Hand opwaarts/ neerwaarts (dimming)
- b. Hand dat voorbij beweegt

Dit zijn zeer simpele handbewegingen. Het is de bedoeling om zo simpel mogelijk te beginnen. Hierdoor worden de onderstaande analyses makkelijker uitvoerbaar.

In een latere fase zal er gekeken en geëxperimenteerd worden met complexere handbewegingen.

- **Cover glas analyse**

Om de opstelling te gebruiken in het werkveld is het noodzakelijk dat de ToF-sensoren achter een doorschijnende materie geplaatst kunnen worden. Dit zal de ToF-sensor beïnvloeden. We moeten deze afwijking dus corrigeren in de geschreven software en/of een ToF-sensor kiezen die hiertegen kan.

- **Omgeving licht analyse met cover glas**

De opstelling zal ook op verschillende momenten en plaatsen gebruikt kunnen worden (denk aan een zonnige dag op een festival, in een werkplaats dat belicht wordt met kunstlicht, ...). Deze interferenties zullen de metingen beïnvloeden en moeten we eveneens in de software corrigeren en/of een ToF-sensor kiezen die hiertegen kan.

- **Simpele handbewegingen met 3 sensoren**

De complexiteit van metingen wordt hier verhoogt. Wanneer we meerdere bewegingen willen detecteren is er ook nood aan meerdere sensoren die metingen uitvoeren.

- a. Hand dat voorbij beweegt (links, rechts, boven, onder)

Niet in Scope

- **Interface**

De gemaakte applicatie zal enkel instaan om de (hand) bewegingen te herkennen en hiervoor een respectievelijk commando voor voorzien en uitsturen. De applicatie waar deze extra gebruikersinterface wordt aan toegevoegd moet ook met de gemaakte applicatie kunnen communiceren a.d.h.v. een zelfgemaakte API (library/ driver).

- **Complexere handbewegingen herkennen met 2 hand**



← v →

De simpele handbewegingen blijven we behouden, maar meerdere commando's zullen ervoor moeten toegevoegd worden. Zo zal er bij het zoomen, ook tabs zijn om een drukknop te simuleren i.p.v. in- en uit te zoomen (bij tabs zal v in "Figuur 4 – tab vs zoom" sneller verlopen dan zooming).

Figuur 4 - tab vs zoom

- **Low Power**

De gehele schakeling op low power brengen zodat dit zeer weinig verbruikt wanneer er niets gebeurt. Zo kan dit langdurig bijvoorbeeld op een batterij werken. Een must voor een IoT device.

- **Productie klaar product uitwerken**

Hiermee wordt bedoeld dat het ontwikkelde product klaar is voor de markt om gebruikt te worden.

Planning

Om dit project tot goed einde te brengen is het vanzelfsprekend dat hierachter een duidelijke en gestructureerde planning moet zijn.

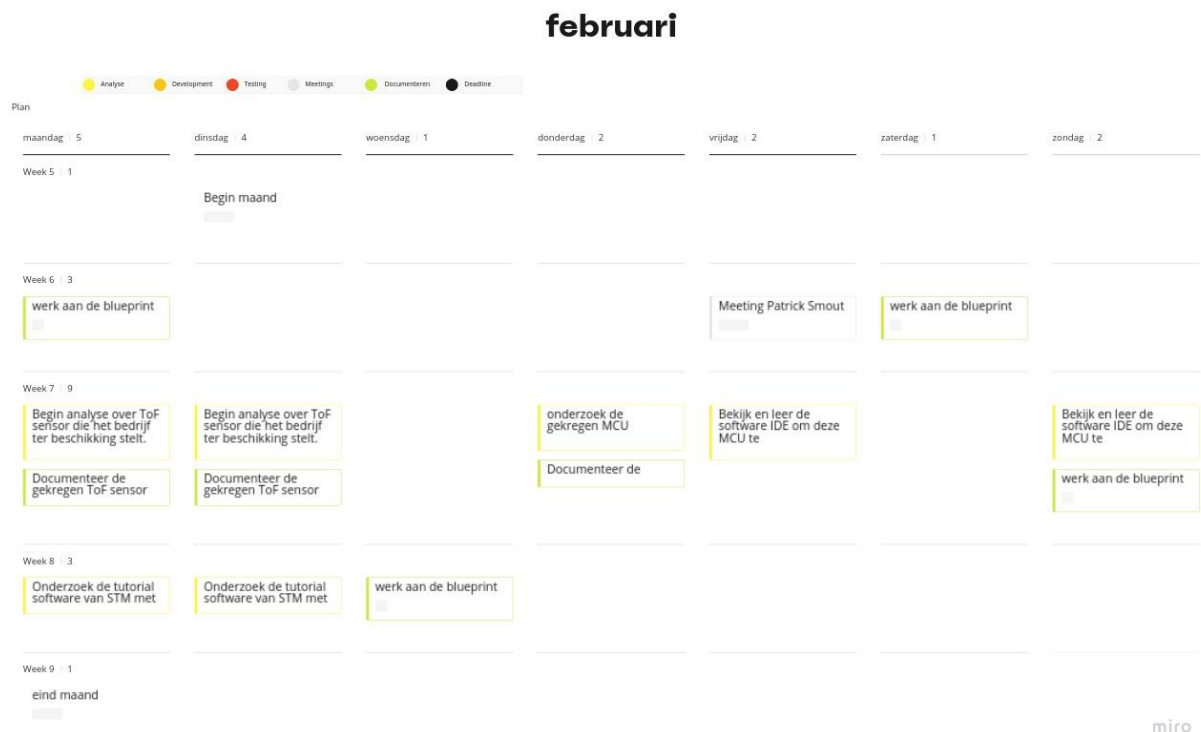
De methodiek die in dit project gebruikt wordt is Kanban dat via het Agile principe werkt. Het Kanban-board maken en onderhouden wordt gedaan in Miro. ([4])

Hierin wordt een maandplanner verwerkt die ingevuld wordt met de aangemaakte taken. Naast de maandplanner wordt er ook een Kanban board voorzien die gebruikt wordt om story's in te plaatsen. (zie "Figuur 5 – voorbeeld: Kanban board maand" en "Figuur 6 – voorbeeld: story's "). Zo kunnen zowel de stagebegeleider als stage mentor zien wat er wanneer gebeurt, en waarmee we met wat bezig zijn.

De update aan de stagebegeleider zal telkens aan het begin van elke week zijn waar beknopt samengevat wordt wat vorige week gepresteerd is, wat we deze week willen bereiken en wat er gepland staat om volgende week te doen.

Wanneer bepaalde zaken niet verwezenlijkt werden in de desbetreffende week, worden deze naar de volgende week verschoven.

Dit mag natuurlijk geen gewoonte worden. Hier zal zeker een iteratief proces over lopen om goed te kunnen inschatten hoeveel tijd bepaalde zaken vragen. Als we alle zaken van de desbetreffende week al af hebben, dan kunnen er extra zaken bijkomen die in de volgende weken gepland stonden.



Figuur 5 - voorbeeld: Kanban board maand

Backlog | 21

In progress | 2

Done | 0

Als user wil ik een commando koppelen wanneer ik mijn hand van boven naar onder beweeg.

Als user wil ik een commando koppelen wanneer ik mijn hand van onder naar boven beweeg

Als user wil ik een commando koppelen wanneer ik mijn hand van links naar rechts beweeg

Als user wil ik een commando koppelen wanneer ik mijn hand van rechts naar links beweeg

Als developer wil ik een duidelijke documentatie die mij opweg kan helpen om de geschreven software te gebruiken

Als user wil ik voorbij de sensoren wandelen zonder dat er iets gebeurt

Als developer wil ik meerdere testings uitvoeren aangaande de gestures. Dit vooraleer de effectieve code geschreven wordt.

Als developer wil ik een gemakkelijk bruikbaar testing board hebben dat ik kan verbinden met headers op een Arduino Uno layout board.

Figuur 6 - voorbeeld: story's

Enkele deadlines over de documentatie:

1. **Blueprint** (analyse) inleveren op **6/3/2022**,
2. **Testplan** inleveren op **3/4/2022**,
3. **Testresultaten** inleveren op **15/5/2022**,
4. **Documentatie en finaal document** op **31/5/2022**

(Finaal document bestaat uit 3 bovenstaande documenten met inhoudstabel, voorwoord en slot).

De **maand februari** plannen wordt ingepland als opstart maand van het project. Omdat er vele componenten ter beschikking zijn, zal er veel onderzoek worden gedaan. Zo selecteren we diegenen die in ons project het beste gaan presteren. Omdat sommige software en hardware onbekend is, gaan we onszelf hierin verdiepen zodat er optimaal aan het project gestart kan worden. Omdat deze maand ons een tijdsspanne geeft van 3 weken, zal het mogelijk zijn om al een 1^{ste} PCB te ontwikkelen om later mee te experimenteren.

De deadlines voor eind februari:

1. Ken de eigenschappen van de gekozen ToF-sensor
2. Ken eigenschappen van de gekozen MCU
3. Ken de software STM32CubeIDE of alternatieven en kan deze gebruiken
4. Vooruitgang/ af hebben blueprint document
5. Eerste draft PCB V0.1

In de **maand maart** zal, zoals hierboven beschreven, de blueprint ingeleverd moeten worden. Deze maand starten we ook met het ontwikkelen van onze applicatie en het documenteren van het testplan. De reeds getekende PCB wordt hier ook besteld en bestukt (al dan niet extern). Parallel met deze gegevens gaan we ook een opstellingV0.1 voorzien om het huidige prototypen in op te stellen om verdere testing waar te maken.

De deadlines voor eind maart:

1. Detecteer aanwezigheid
2. Detecteer eenvoudige gebaren/ complexere gebaren
3. Vooruitgang testplan document
4. Ingebruikname PCBV0.1
5. Eerste draft opstellingV0.1

De **maand april** is de maand waarin het testplan moeten inlevert worden. In deze maand is het ook de bedoeling om een stap verder te gaan met het project. De analyse van het cover glas en de interferentie van licht worden hierin ontwikkelt. Het testplan dat we voorop hebben opgesteld, zal in deze maand ook uitgevoerd worden op de gemaakte schakeling. De resultaten komen in een apart testresultaten document. We hebben in deze maand ook de mogelijkheid om een PCBV0.2 te maken indien dat PCBV0.1 niet voldoet aan de verwachtingen.

De deadlines voor eind april

1. Opheffing cover glas interferentie
2. Opheffing interferentie licht
3. De applicatie testen volgens opgesteld testplan
4. Vooruitgang testresultaten document
5. Eventuele PCBV0.2

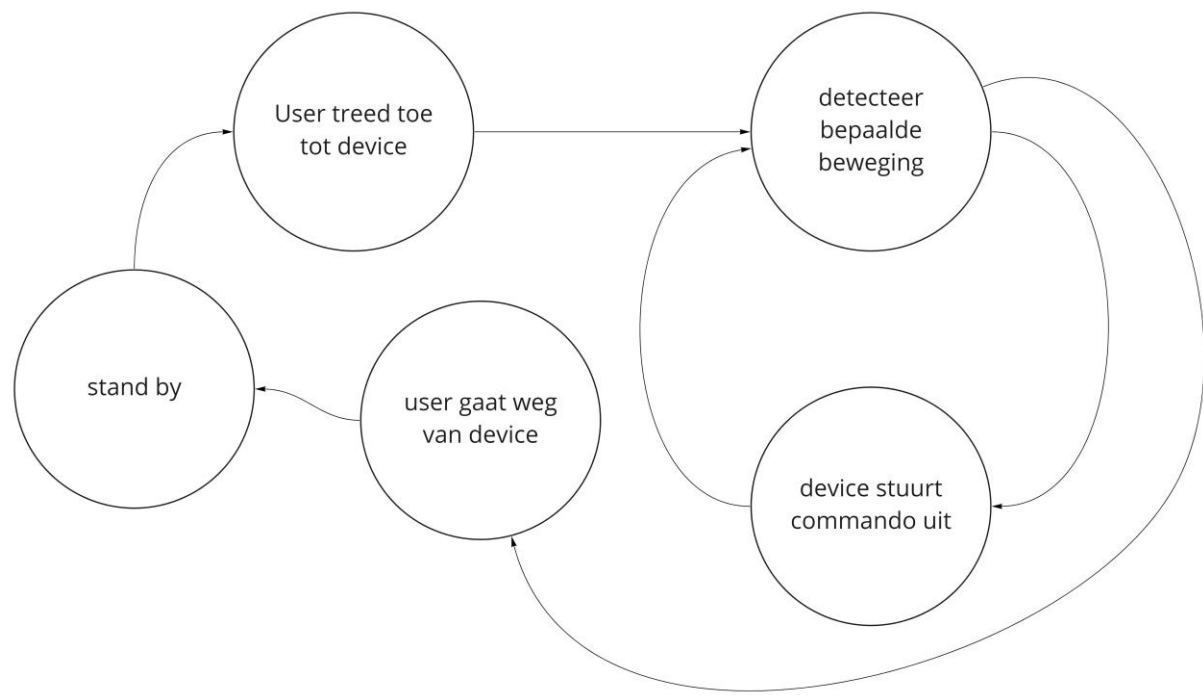
De laatste **maand mei** zal vooral gaan om nog verdere testing en het bestukken van een finaal PCB1.0. Ook maken we hier een finale opstelling (maquette) zodat er een mooi en werkend eindproduct staat. Als alle zaken goed gaan, zal er meer verdiept worden in de "*niet in scope*" opdrachten.

De deadlines van eind mei:

1. Indienen finaal document
2. Indienen documentatie
3. Maquette
4. PCB1.0
5. Eventueel niet in scope opdrachten

Functioneel design

De functionaliteiten van dit project zijn, een extra dimensie aan de gebruiker bieden. De interne flow wordt in “Figuur 7 – functioneel design” weergegeven.



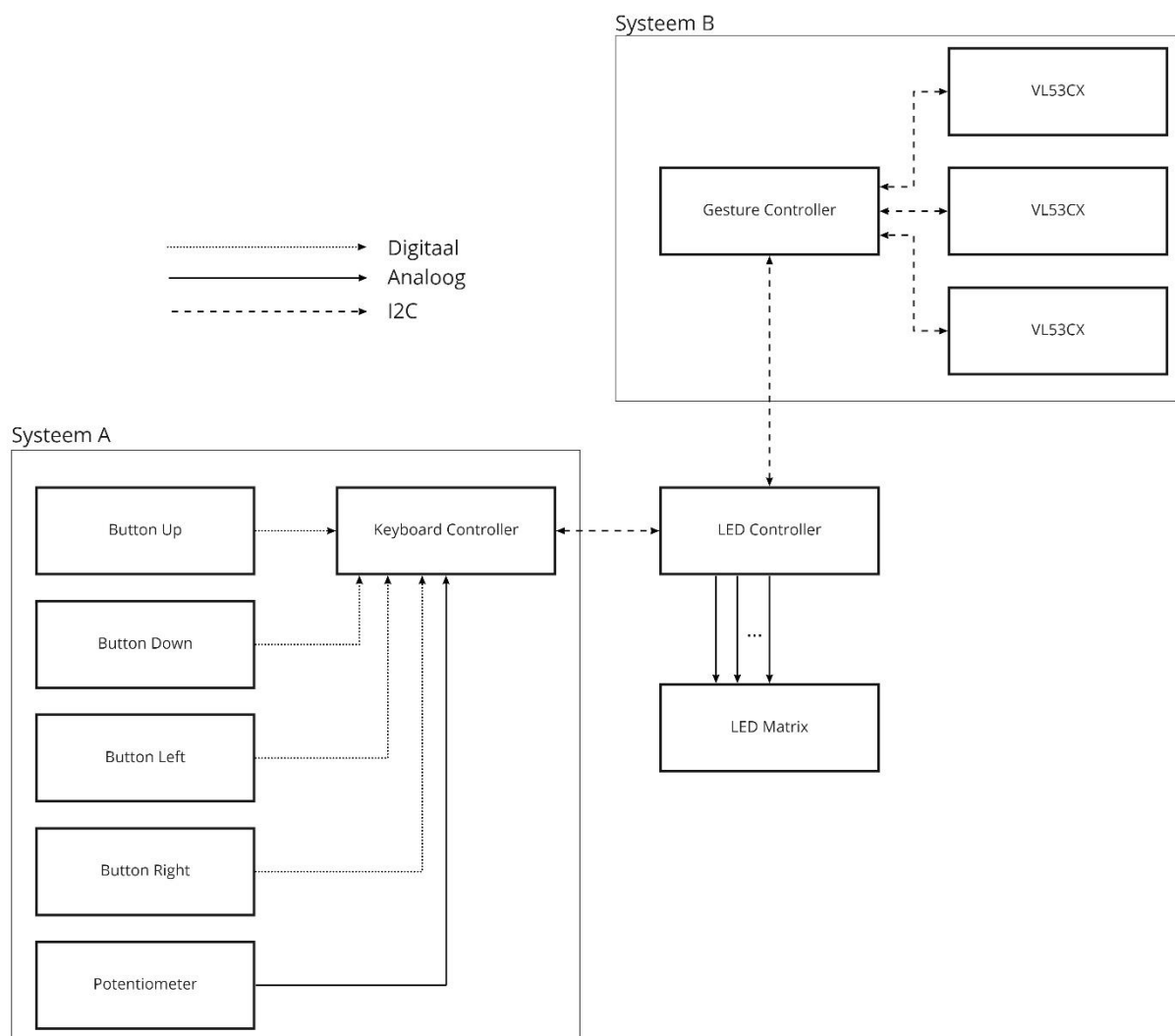
Figuur 7 - functioneel design

Technisch design

Hardware analyse

In deze paragraaf zal er verdiept worden in de hardware zijden van het project.

“Figuur 8 - smart objects” geeft een duidelijk beeld van wat we nu juist gaan bouwen. We gaan een extensie geven voor de gebruiker zijn toegankelijkheid aan een systeem. Hierdoor kan de Keyboard Controller (Systeem A) eventueel vervangen worden door de Gesture Controller (Systeem B). Beide systemen zorgen namelijk voor dat de gebruiker kan interacteren met de LED Controller. Deze zal op zijn beurt de gekregen informatie verwerken en dit weergeven in de LED Matrix.



Figuur 8 - smart objects

We kunnen nu dus Systeem B eender waar implementeren waar men een interface heeft voorzien (mits er rekening houden word met het aantal commando's die mogelijk zijn met de Gesture Controller).

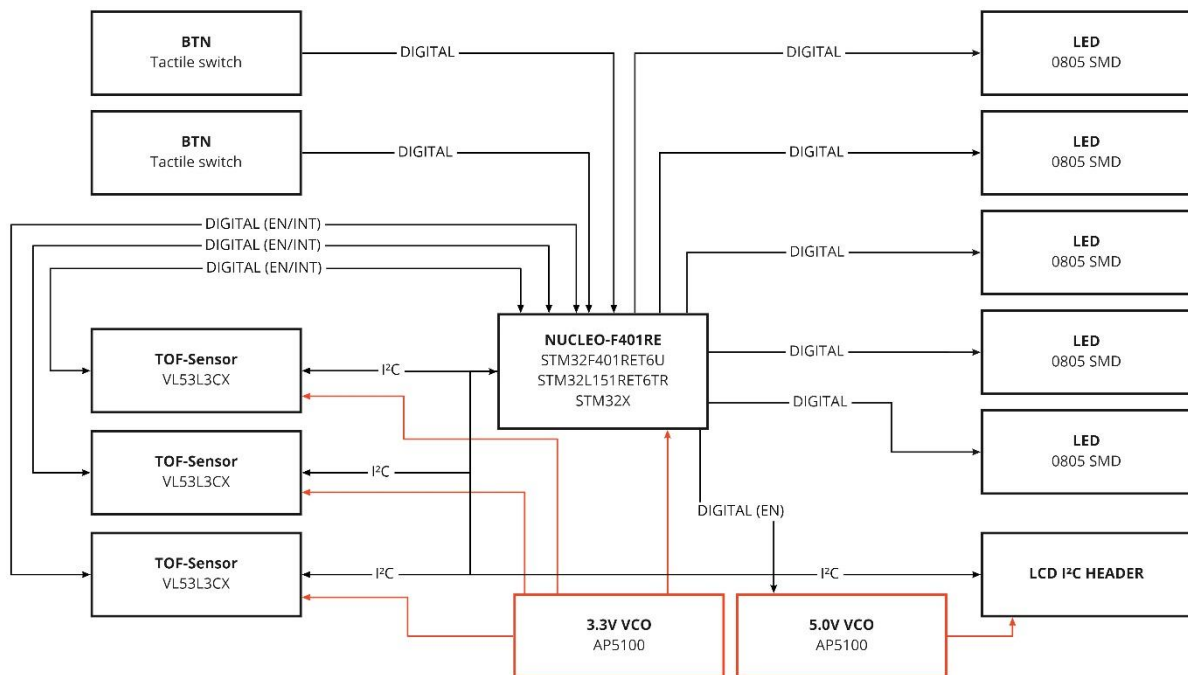
Omdat er in latere fase zelf een PCB wordt ontwikkelen is het van belang om hiervoor eerst grondige studie te doen van wat we allemaal nodig hebben. In deze sectie is het high level schema “*Figuur 9 – blokdiagram*” en enkele karakteristieken opsommen van de gebruikte blokken te zien.

Blokdiagram

In onderstaand schema “*Figuur 9 - blokdiagram*” is te zien dat er gebruik gemaakt word van 3 ToF sensoren, 1MCU, 5 Light Emitting Diodes’s (LED’s), 2 drukknoppen (BTN’s) en Voltage Regulators (VR’s).

De 5 LED’s en 2 BTN’s zijn enkel en alleen om in de analyse fase ermee te kunnen spelen. Bij het eindproduct is het de bedoeling dat de commando’s die we uitzenden, verwerkt worden door een extern systeem zoals in “*Figuur 8 – smart objects*”.

De MCU is zo gekozen omdat deze op het evaluatie bordje staat. Doordat op de huidige moment de voorraad van componenten zeer beperkt is, gaan we ook kijken voor andere gelijkaardige MCU’s. Dit gaan we voorzien op onze pcb zodat hiervoor enkel wat solder pads verlegd moeten worden om een andere MCU te gebruiken.



Figuur 9 - blokdiagram

Specificaties

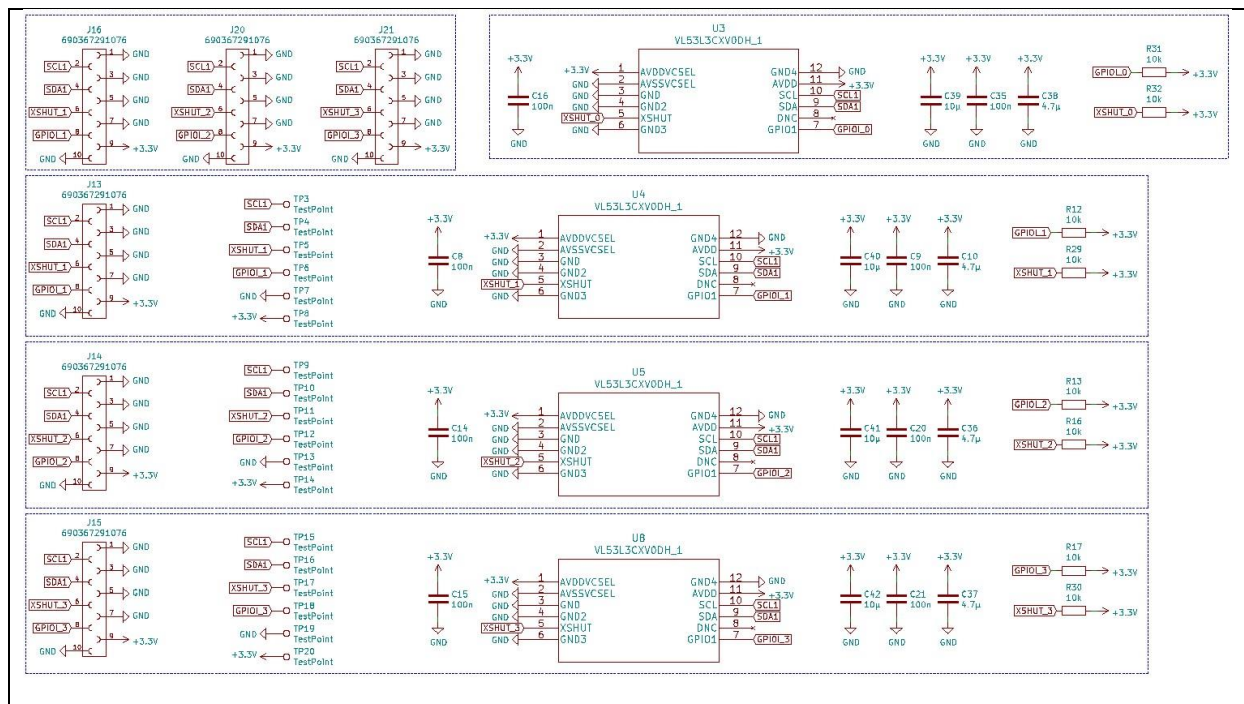
In volgende tabel bespreken we blok per blok (dat te zien is in “Figuur 9 – blokdiagram”) de belangrijkste eigenschappen gevonden in de datasheets.

Blok	Specificatie	Minimum	Nominaal	Maximum	Eenheid
ToF-sensor VL53L3CX	Spanning	2.6	2.8	3.6	V
	Stroom	0.003	16	18	mA
	I ² C klok frequentie	0	-	1000	MHz
	Adres	0x00	0x52	0xFF	hex
	Afstand	-	-	3	m
	Field of View (FoV)	-	25	-	°
NUCLEO-F401RE STM32F401xE	Spanning	1.7	3.3	3.6	V
	Stroom	3.8	-	160	mA
	CPU-frequentie	4	-	84	MHz
	Aantal GPIO's	-	48	-	-
NUCLEO-F401RE STM32L151RET6TR	Spanning	1.65	-	3.6	V
	Stroom	0.042	-	100	mA
	CPU-frequentie	0.032	-	32	MHz
	Aantal GPIO's	-	47	-	-
3.3V/ 5V VR AP5100	Ingang spanning	4.75	-	24	V
	Uitgang spanning	0.81	-	15	V
	Stroom	0.01	-	1200	mA
	Efficiëntie	60	85	90	%
	Frequentie	-	1.4	-	MHz
LED 0805 SM0805URC	Stroom	22	30	100	mA
LCD I ² C HEADER	Spanning	3.3	5	5	V
	adres	0x20	0x27	0x27	hex

Onderliggende argumentatie

Omdat er voor sommige componenten wel meer dan 100 soorten bestaan, is het van belang om ze allemaal te bekijken en hieruit de beste en meest beschikbare te selecteren. Hieronder is de argumentatie die de genomen component zal staven.

BLOK	ARGUMENTATIE	ALTERNATIEVEN
ToF-sensor VL53L3CX	De VL53L3CX is de nieuwste Time-of-Flight (ToF) product van STMicroelectronics en integreert ST's derde generatie FlightSense gepatenteerde technologie. Het combineert een hoogwaardige nabijheids- en afstandssensor met afstandsmetingen voor meerdere objecten en automatische veeg correctie. Hiervan is ook een evaluatie bord ter beschikking.	<ul style="list-style-type: none">• VL53L1• VL53L0X• VL53L1X
NUCLEO-F401RE STM32F401xE	De MCU die we in deze opstelling gaan gebruiken is een 32-bit MCU van de STM32F reeks. Het is een zeer robuuste MCU die vele mogelijkheden aanboort heeft om te gebruiken. Het wordt ook meegeleverd met een evaluatie bord samen met de ToF-sensor. Deze beschikt over Arduino compatibele headers om zo gemakkelijk Arduino shields te gebruiken.	<ul style="list-style-type: none">• NUCLEO-F446RE• NUCLEO-F411RE• NUCLEO-L452RE• STM32L151xE
3.3V/ 5V VR AP5100	Om de zaak te voeden, is er nood aan een component die ons de nodige spanningen voorziet. We gaan gebruik maken van een geschakelde voeding (buck converter) omdat dit een extra geeft aan de opbouw. Om de uitgangsspanning te regelen moeten we een feedback spanning meegeven door 2 weerstanden. We kunnen dus 1 IC gebruiken voor de 2 spanningen.	<ul style="list-style-type: none">• L6981• LM2596HV• LM1117• AMS1117
LED SM0805URC	Wanneer de MCU bepaalde acties uitvoert is het vanzelfsprekend dat de MCU dit kenbaar maakt zodat we in de analyse fase snel en efficiënt kunnen testen.	<ul style="list-style-type: none">• TC-9201956• LYR976-PS-36-Z
LCD I ² C HEADER	We gaan hiermee makkelijker kunnen zien wat de MCU nu juist interpreteert als commando. We nemen een header zodat dit later eraf gehaald kan worden.	<ul style="list-style-type: none">• OLED I²C• OLED SPI
BTN tactile switch	Om eventuele testen uit te voeren is soms een knopje noodzakelijk om zaken te simuleren. Hier hebben we drukknoppen genomen die courant gebruikt worden.	<ul style="list-style-type: none">• Toggle switch• Dip switch



In bovenstaande schema's is te zien hoe we de zaak trachten aan te sluiten als 1 geheel. Deze is nog niet definitief en hier zal er nog een iteratief proces over lopen. De hoofd delen zijn hierin al wel verwerkt zoals in "Figuur 9 – blokdiagram" te zien zijn.

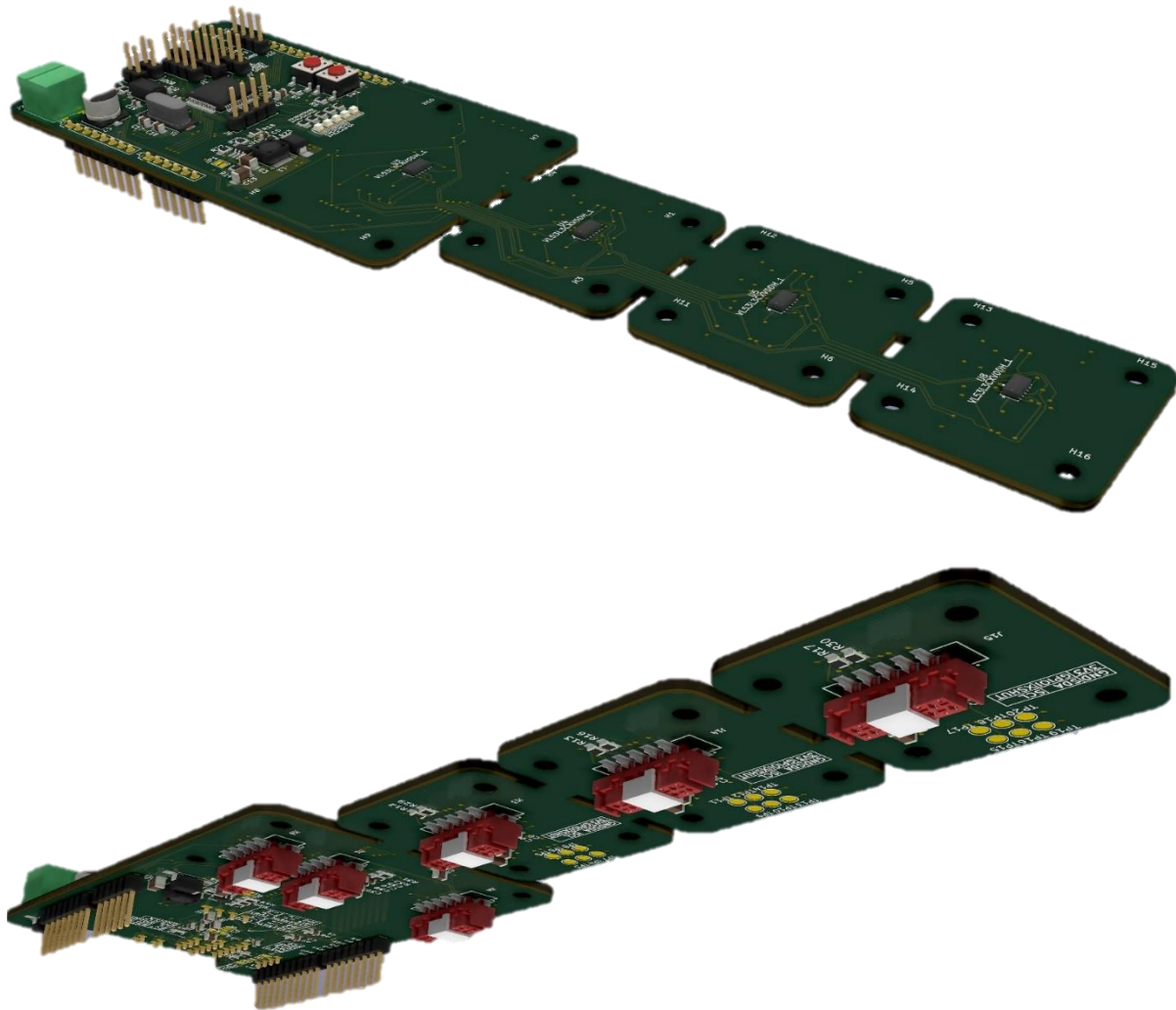
In de 2^{de} figuur is te zien dat er 2 MCU's staan. Dit is zo gedaan omdat er dan gekozen kan worden tussen 2 verschillende MCU's.

De laatste foto is het schema van de VL53L3CX sensoren. We hebben er 4 gekozen zodat we nog kunnen experimenteren met een 4^{de} ToF-sensor. We voorzien ook connectoren om een grotere afstand tussen de sensoren te voorzien.

Wat ook niet onbelangrijk is, is dat we de pinheaders op zo'n wijze willen construeren dat deze de Arduino Uno formaat bevatten. Op deze manier kan er makkelijk tussen verschillende platformen gewerkt worden. Hierdoor is het gemaakte bordje ook makkelijk bruikbaar op de gekozen Nucleo-F401RE die Arduino Uno compatible headers heeft. "Figuur 10 – 3D PCB V0.1" geeft een zicht op wat we willen bereiken.

Omdat dit een analyse bordje is, zullen er ook meerdere componenten voorzien worden die in latere fase nog aangesproken kunnen worden.

Ook is te zien dat we de aparte sensoren kunnen afbreken van het main bordje. Zo kunnen we een grotere afstand bekomen om te testen.



Figuur 10 - 3D PCB V0.1

Software Analyse

Wanneer de hardware kant vervolledigd is, moeten we hier uiteraard software in plaatsen zodat de schakeling kan werken. Het is belangrijk dat er, vooraleer we starten aan het project, alles uitgezocht word hoe en wat we waar gaan doen aangaande software.

Data in/ Out

In onderstaande tabel is kort geschreven wat er in en uit de blokken ("*Figuur 9 blokdiagram*") komt.

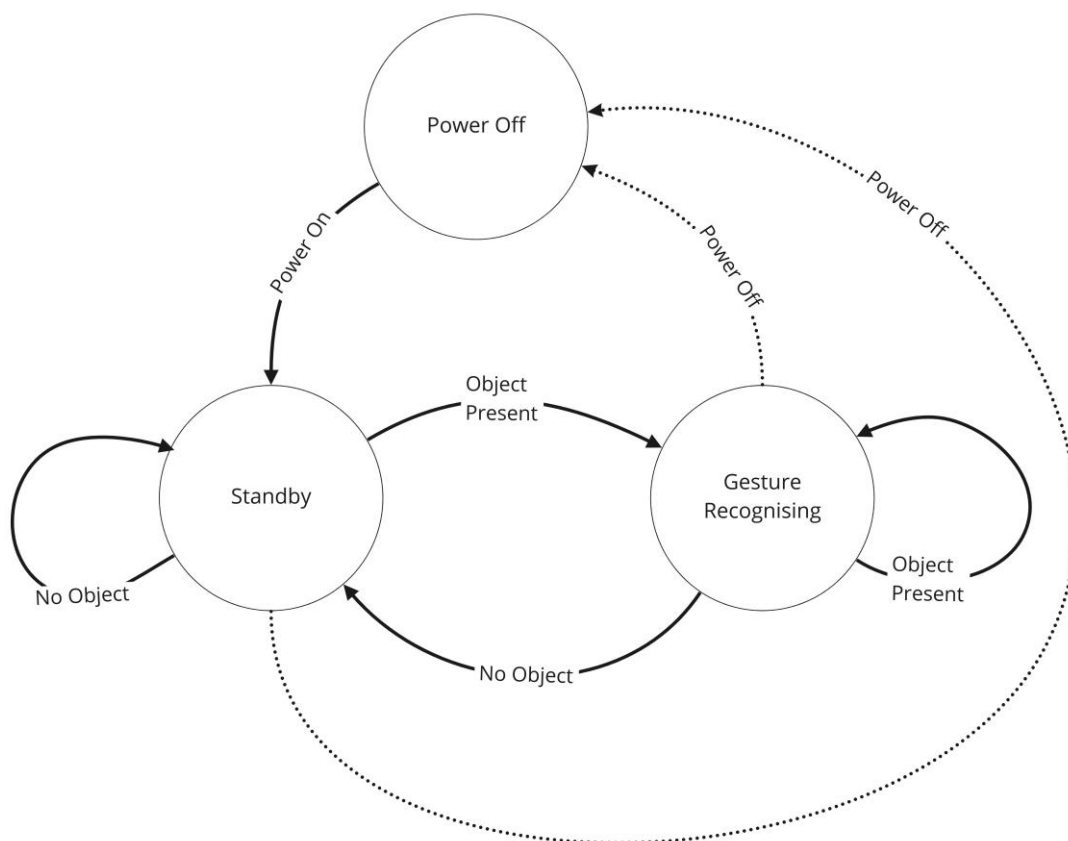
Blok	Data In	Data Uit
ToF-sensor VL53L3CX	Configuratie instellingen	Sensordata, interrupt
NUCLEO-F401RE STM32F401xE	Configuratie instellingen, Sensordata	Commando's, Configuratie instellingen, Sensordata
3.3V/ 5V VCO AP5100	Digital "1"/ "0"	-
LED SM0805URC	Digital "1"/ "0"	-
LCD I ² C HEADER	Display data	-
BTN Tactile	-	Digital "1"/ "0"

State-diagram

Het programma zal een aantal statussen bevatten die de high level flow wat meer verduidelijken.

Zo is te zien in “Figuur 11 – state-diagram” dat, wanneer er power is, we in Standby staan. Deze state blijft als er geen object te zien is.

Wanneer er een object zichzelf presenteert, dan wordt er naar een andere state genaamd Gesture Recognising gegaan waar we de eventuele gestures zullen herkennen. Deze state blijft enkel aan als er een object aanwezig is. Zo niet, gaan we terug naar de Standby state. In elke state waar we ons in bevinden kan ten allentijden de voeding wegvallen wat ons weer naar de Power Off state brengt.

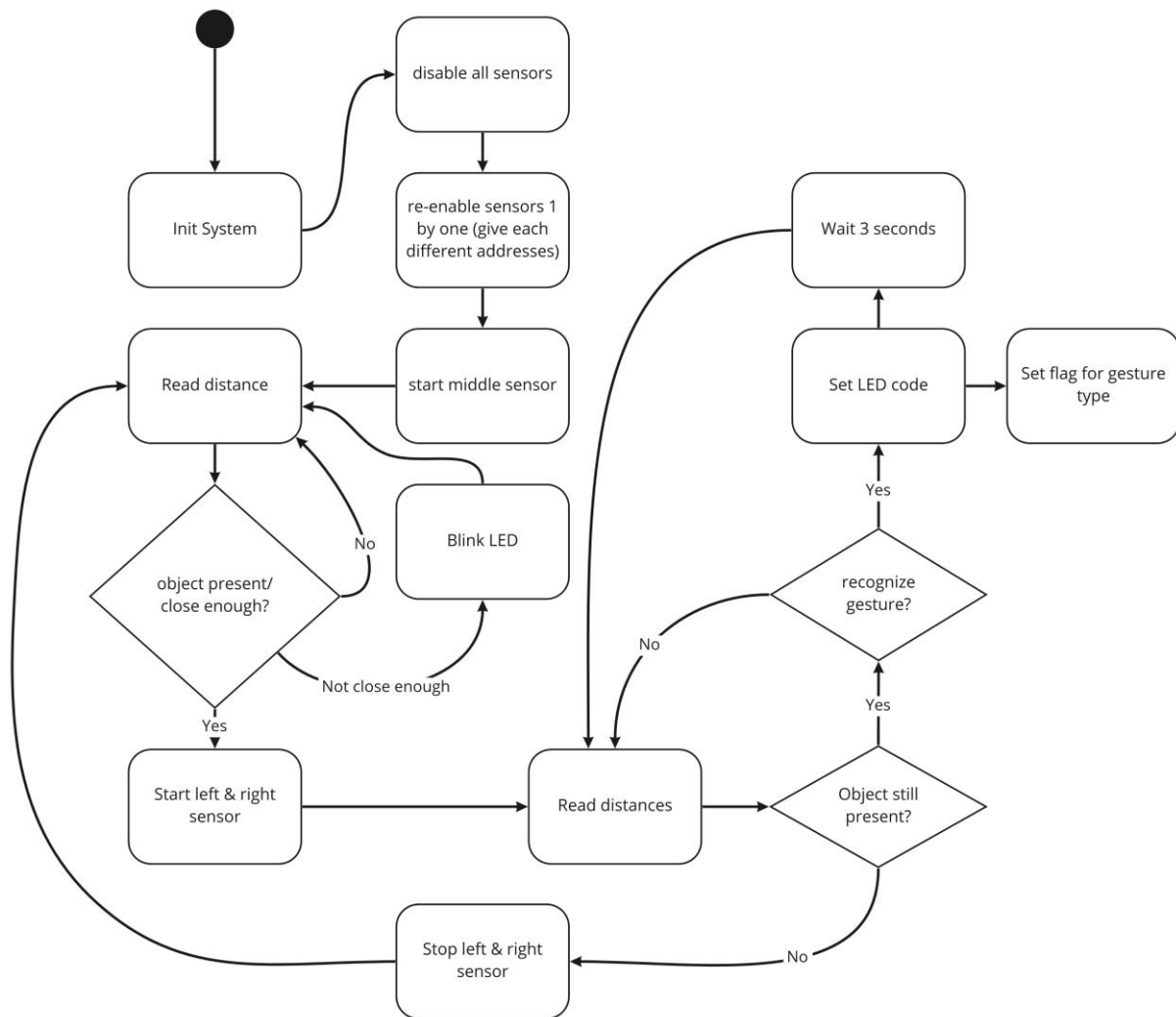


Figuur 11 – state-diagram

Flowchart

De gemaakte flowchart (te zien op “*Figuur 12 - flowchart*”) zal op een dieper level de software beschrijven. Deze kan uiteraard nog wijzigen bij volgende iteraties van de software. Voor initiële start van het project gaan we deze flowchart aannemen als leidraad.

De 2 states, die ook op “*Figuur 11 – state-diagram*” staan, staan hier ook te zien zijn in figuur “*Figuur 12- flowchart*”. De flowchart legt zichzelf uit.



Figuur 12 - flowchart

Beschrijving van de mogelijke interfaces

Bij het hoofdstuk *Project definitie* bij het deel *niet in scope* is er gesproken over de interface die we eventueel zullen maken wanneer de *scope* opdrachten voltooid zijn. De interface is dus een extra aan dit project.

Omdat er gewerkt wordt met een MCU kunnen we deze verschillende interfaces geven. Ook is het mogelijk om verschillende communicatiemiddelen te gebruiken om de commando's over te brengen (zoals Inter-Integrated Circuit (I²C), Serial Peripheral Interface (SPI), Universal Asynchronous Receiver-Transmitter (UART), Universal Synchronous Asynchronous Receiver Transmitter (USART), ...).

Omdat de gekozen MCU 3 I²C interfaces bevat, (waarvan we met huidig General Purpose Input/Output (GPIO) gebruik 2 van kunnen gebruiken) is dit het communicatiemiddel waarmee de commando's worden verzonden en ontvangen. De gemaakte opstelling zal als slave worden opgesteld en zal dus wachten totdat de master hier iets naar stuurt.

De communicatie zal dus verlopen dat de master het commando X verstuurd naar de slave en de slave zal met Y antwoorden. We gaan hier 1 byte voor gebruiken waarmee 255 commando's kunnen geconstrueerd worden. De commando's wordt gedefinieerd in hexadecimaal zodat dit duidelijk is. In de code zal er gebruik gemaakt worden van `uint8_t` types.

Net zoals bij de ToF-sensoren gaan wij ook beschikken over een hardware interrupt pin die als doel heeft de master te waarschuwen wanneer de slave iets heeft waargenomen. We kunnen ook alles op softwareniveau doen.

In volgende tabel worden de commando's verduidelijkt. Deze lijst kan achteraf nog uitbreiden afhankelijk van de aantal gestures en functionaliteiten van ons product.

Toestel	Commando	Uitleg																																
master	0x01	Opvraging commando																																
	0x10	Geen object aanwezig																																
	0x20	Een object aanwezig																																
	0x21	Er is in de afgelopen 3 seconden een gesture geweest die van links naar rechts ging.																																
	0x22	Er is in de afgelopen 3 seconden een gesture geweest die van rechts naar links ging.																																
	0x23	Er is in de afgelopen 3 seconden een gesture geweest die van boven naar onder ging.																																
	0x24	Er is in de afgelopen 3 seconden een gesture geweest die van onder naar boven ging.																																
slave	0x25	Er is een hand op en neer aan het bewegen.																																
	Dit wordt gevolgd door een waarde van 1024.																																	
10-bit verdeeld als:																																		
1 ^{ste} byte heeft 8 eerste bits beginnend van Least-significant bit (LSB)																																		
2 ^{de} byte die de overige 2 bits bevat tot Most-significant bit (MSB)																																		
<table><tr><td colspan="8">2^{de} byte</td><td colspan="8">1^{ste} byte</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>MSB</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>LSB</td></tr></table>			2 ^{de} byte								1 ^{ste} byte								0	0	0	0	0	0	MSB	x	x	x	x	x	x	x	x	LSB
2 ^{de} byte								1 ^{ste} byte																										
0	0	0	0	0	0	MSB	x	x	x	x	x	x	x	x	LSB																			
0x26																																		

Beschrijving van eventuele impact op de huidige infrastructuur

De huidige (test) infrastructuur bezit momenteel enkel knoppen. Het doel van dit project is om gesture aan de bestaande infrastructuur toe te voegen en in beste geval de knoppen vervangen door de gesture.

Hier zal een migratie voor nodig zijn omdat de meeste knop-interfaces met digitale ingangen werken en ons product werkt met I²C. in een latere fase kan ons product ook meerdere interfaces bevatten zoals digitale uitgangen die de gestures voorstellen en/ of communicatie via SPI i.p.v. I²C.

Bij IoT toepassingen, zien we wel dat meeste toestellen standaard I²C aanboort hebben. Hiervoor zal er weinig of zelfs geen migratie voor moeten gebeuren.

Documentatie

Tijdens het schrijven van de code gaat commentaar voorzien worden bij elk belangrijk deel. Dit zal tijdens de development belangrijk zijn voor mezelf zodat we een overzicht houden van de al reeds geschreven code. Tijdens de aanvullingen van de code, is het ook de bedoeling dat we deze refactoreren met bijhorende commentaar.

We gaan als eind documentatie voor het bedrijf een technische documentatie voorzien zowel voor de software als de hardware.

Voor de software moet dit een document zijn waarin alles beschreven staat wat men juist nodig heeft om van scratch het project terug op te bouwen en werkend te krijgen. Hierin zullen zaken staan zoals de gebruikte IDE, de versie van de compiler, de gebruikte bibliotheken, ...

Aan de Hardware kant zal er voor ons eind PCB de nodige documentatie voorzien worden zoals bijvoorbeeld testresultaten, Gerber bestanden, 3 Dimensionele (3D) tekeningen, ...

Voor de opleiding moet er een testplan (03/04) en testresultaat (15/5) document opgeleverd worden. Hiervan zullen we zeker zaken gebruiken in ons eind documentatie.

Bronvermelding

- [1] Time-of-Flight principle. (2021, 22 april). In Terabee. <https://www.terabee.com/time-of-flight-principle/>
- [2] USART vs UART: Know the difference. (2015, 21 september). In EDN <https://www.edn.com/usart-vs-uart-know-the-difference/>
- [3] Custom-made elektronica & besturingen | E.D.&A. (z.d.). ED&A. Geraadpleegd op 22 januari 2022, van <https://www.edna.eu/nl>
- [4] vanderwardt@agilescrumgroup.nl. (2020, 24 december). Wat is Kanban? Een volledige uitleg van de Kanban Methode + template. Agile Scrum Group. Geraadpleegd op 7 februari 2022, van <https://agilescrumgroup.nl/wat-is-kanban-methode/>