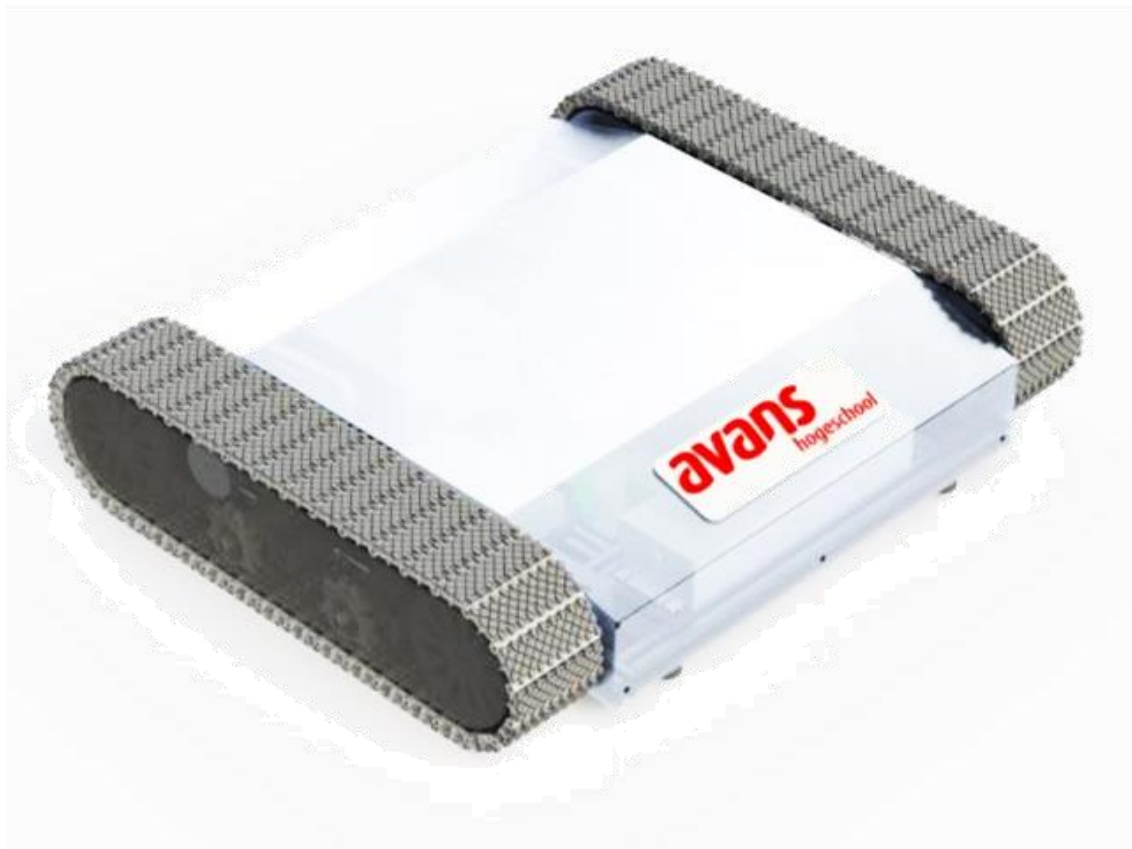


Ontwerprapport inspectierobot V2



Robert Franken, 2109133

Thom van Damme, 2112137

Thomas Brouwers, 2118820

Ewout de Gast, 2123991

Datum: 27-10-2019

Ontwerprapport inspectierobot V2

Auteurs:

Robert Franken, 2109133

Thom van Damme, 2112137

Thomas Brouwers, 2118820

Ewout de Gast, 2123991

In opdracht van: Avans Hogeschool Breda

Datum: 27-10-2019

Voorwoord

Dit rapport is ontstaan als naslagwerk voor de inspectierobot. In het rapport zal volgens de engineering methode gewerkt worden waardoor de bevindingen en ontwikkelingen van de robot terug te vinden zijn voor eventuele opvolgers van het project. Dit rapport is het tweede rapport voor de inspectierobot. Hoe de volledige robot tot stand is gekomen is in het eerste rapport te vinden.

De inspectierobot is onderdeel van een groter project maar kan wel als onafhankelijk systeem worden gezien. De enige investeerder en opdrachtgever voor het project is Avans Hogeschool Breda.

Het rapport kan door eventuele opvolgers gebruikt worden als informatiebron over de ontwikkeling van de inspectierobot.

Het projectteam is dankbaar om de kans gekregen te hebben de opdracht van de inspectierobot aan te mogen gaan. Het team is Avans Hogeschool dankbaar voor het sponsoren van de materialen. Alle teamleden willen Jos van Kreij bedanken voor het begeleiden van het projectteam.

Breda, oktober 2019

Robert Franken, Thomas Brouwers, Ewout de Gast en Thom van Damme

Samenvatting

Dit verslag volgt het proces van het onderzoek en de realisatie van de inspectierobot voor de Smart Tooling projectgroep. Het doel van de inspectierobot is het inspecteren van olietanks voor vuile plekken die schoongemaakt moeten worden. In het verslag komt de analyse, ontwerp, realisatie, beheer en uiteindelijk de conclusie en aanbevelingen naar voren.

Voor het project is de volgende onderzoeksvraag geformuleerd: Welke aanpassingen zijn benodigd om de huidige inspectierobot operationeel te krijgen om een tankwand te inspecteren op olieresten?

Bij de analyse zijn de bestaande onderdelen van de robot onderzocht op werking en welke aanpassingen noodzakelijk zijn om de robot operationeel te maken. Hieruit is gebleken dat de robot verschillende functionaliteiten mist. Er is geconstateerd dat de Raspberry Pi, Raspberry Shield, aandrijving en ruwheidsensor aangepast dienen te worden. De software voor de robot is opnieuw geschreven.

In het ontwerp zijn er nog een paar andere functionaliteiten toegevoegd om aan de doelstellingen te voldoen. De robot is uitgebreid met een gyrosensor en lidar om de positie in een tank te kunnen bepalen. Er is ook een client interface ontworpen om de robot gebruiksvriendelijker te maken.

Bij de realisatie zijn de ontworpen aanpassingen en uitbreidingen gerealiseerd. De robot gebruikt ROS als framework met Ubuntu als besturingssysteem. De kern van de robot is de NUC. De NUC verwerkt de gegevens van de robot en is ook het toegangspunt voor de client interface. De Raspberry Pi is de link tussen alle sensoren en wordt dus gebruikt als een I/O bord. Voor het besturen van de robot wordt een Xbox controller gebruikt. Hiermee kan de robot handmatig bestuurd worden, de blacklights aangestuurd worden en foto's gemaakt worden.

De robot mist nog een paar functionaliteiten. Deze zullen later nog toegevoegd moeten worden om volledig aan de eisen te voldoen. Hieronder valt de ruwheidsensor en het autonoom rijden.

Inhoudsopgave

VOORWOORD	2
SAMENVATTING	3
INHOUDSOPGAVE	4
1. INLEIDING	6
PROJECT OMSCHRIJVING.....	7
1.1 DOELSTELLINGEN	7
1.1.1 PROGRAMMA VAN EISEN	7
2. ANALYSE VAN DE INSPECTIEROBOT.....	8
2.1 RASPBERRY PI	9
2.2 RASPBERRY PI SHIELD PCB.....	9
2.3 AANDRIJVING.....	10
2.3.1 MOTORDRIVERS	11
2.3.2 BESTURING VIA CONTROLLER.....	11
2.4 CAMERA	11
2.4.1 BLACKLIGHT MODULE	12
2.5 RUWHEIDSENSOR	12
2.6 NUC	12
2.7 CONCLUSIE ANALYSE INSPECTIEROBOT	12
CONCLUSIE ANALYSE.....	13
3. ONTWERP	14
3.1 AANPASSING HUIDIGE ONTWERP	14
3.1.1 RASPBERRY PI & RASPBERRY SHIELD	14
3.1.2 MOTORDRIVER GEGEVENS	14
3.1.3 RUWHEIDSENSOR	14
3.1.4 SOFTWARE NUC	15
3.2 UITBREIDING VAN HET ONTWERP	15
3.2.1 UITBREIDING VAN SENSOREN.....	16
3.2.2 CLIENT INTERFACE.....	18
3.2.3 POSITIONERING VAN DE ROBOT.....	18
3.2.4 BESTURING VIA CONTROLLER	19
3.2.5 NIEUWE RASPBERRY PI SHIELD PCB-ONTWERP.....	19
3.3 VERIFICATIE VAN HET ONTWERP	21
4. REALISATIE.....	23
4.1 GEREALISEERDE AANPASSINGEN	23
4.1.1 RUWHEIDSENSOR.....	23
4.1.2 MOTORDRIVER INSTELLINGEN.....	23
4.2 GEREALISEERDE UITBREIDINGEN	24
4.2.1 ROBOT OPERATING SYSTEM.....	24
4.2.2 RASPBERRY PI SHIELD	28
5. BEHEREN VAN DE INSPECTIEROBOT	33
5.1 ROBOT SYSTEEMONDERDELEN.....	33
5.2 HARDWARE TEST	33
5.3 CLEANROBOT FAT	35
5.4 INGEBRUIKNAME ROBOT EN AANBEVELINGEN	35
6. CONCLUSIE & AANBEVELINGEN	35

7. BIBLIOGRAPHY	36
BRONVERMELDING	38
BEGRIPPENLIJST	39
SYMBOLENLIJST	40
BIJLAGE A – PLAN VAN AANPAK	41
BIJLAGE B – TEST PYTHON SCRIPT MOTORAANDRIJVING	54
BIJLAGE C – AANDRIJVING TESTRAPPORTAGE	55
BIJLAGE D – TESTRAPPORTAGE CAMARA EN BLACKLIGHT LEDS	56
BIJLAGE E – TESTRAPPORTAGE RUWHEIDSSENSOR	57
7.1.1 UITGEVOERDE STAPPEN	57
BIJLAGE F – MOTOR BEREKENINGEN	59
BIJLAGE G – RASPBERRY PI SHIELD SCHEMA	61
BIJLAGE H – INSTALLATIE UBUNTU OP NUC	62
BIJLAGE I – INSTALLATIE ROS OP UBUNTU	63
BIJLAGE J – VERBINDING CONTROLLER MET NUC	64
BIJLAGE K – INSTALLATIE DHCP SERVER	65
BIJLAGE L – INSTALLATIE PACKAGES	66
BIJLAGE M – OPSTARTEN ROBOT	67
BIJLAGE O – BEDRADING RUWHEIDSSENSOR	70

1. Inleiding

Voor het schoonmaken van olietanks mogen vanaf 2020 geen mensen meer in de olietanks aanwezig zijn vanwege veiligheidsredenen. Smart Tooling is een clustergroep die zich bezighoudt met een vervanging te zoeken om de olietanks schoon te maken. Als oplossing is ervoor gekozen om een schoonmaakrobot te maken die autonoom de wanden van de olietanks schoonmaakt. De druk op de schoonmaakrobot zal echter te groot zijn voor sensoren om vuile plekken nog te kunnen vinden. Vanwege dit probleem moet er dus een tweede inspectierobot gemaakt worden die dat kan gaan doen.

Vorig jaar is de inspectierobot door een andere groep studenten mechanisch gerealiseerd. De robot mist nog wel functionaliteiten die nodig zijn om een oliewand te kunnen inspecteren. Dit moet onderzocht worden door een groep nieuwe studenten daarvoor is de volgende hoofdvraag opgesteld:

Welke aanpassingen zijn benodigd om de huidige inspectierobot operationeel te krijgen om een tankwand te inspecteren op olieresten?

Om deze vraag te beantwoorden is de huidige staat van de robot en de aanwezige documentatie onderzocht. Het onderzoek richt zich enkel op het verbeteren van de inspectierobot die door de vorige studentengroep is gerealiseerd.

De opbouw van het verslag is als volgt: Eerst wordt het project omschreven, hierbij worden de deelvragen, doelstellingen eisen en het projectresultaat beschreven. In hoofdstuk 2 wordt de analysefase van het project beschreven, hierbij worden de deelsystemen apart geanalyseerd. In hoofdstuk 3 wordt het ontwerp van het product getoond.

Project omschrijving

In dit hoofdstuk worden de projectdoelen, programma van eisen en projectresultaat beschreven.

1.1 Doelstellingen

De doelstellingen van het project worden uit de deelvragen gehaald. Deze deelvragen zijn als volgt:

1. Uit welke elektrische deelsystemen is de robot opgebouwd?
2. Hoe zijn de elektrische deelsystemen aangesloten en welke protocollen worden hiervoor gebruikt?
3. Welke handelingen zijn vereist om een deelsysteem te testen op volledige werking?
4. Welke deelsystemen in de robot moeten verbeterd of vervangen worden?

Met behulp van de doelstellingen kan de hoofdvraag beantwoord worden. De doelstellingen zijn als volgt:

1. Een overzicht hebben waarbij alle deelsystemen en de verbindingen daartussen duidelijk zijn toegelicht.
2. Documentatie van de code die gebruikt is voor de functionaliteiten van de deelsystemen.
3. Een lijst van nieuwe hardware die gebruikt wordt voor de robot.

De werkwijze bij deze deelvragen en doelstellingen staat beschreven in het Plan van Aanpak in bijlage A.

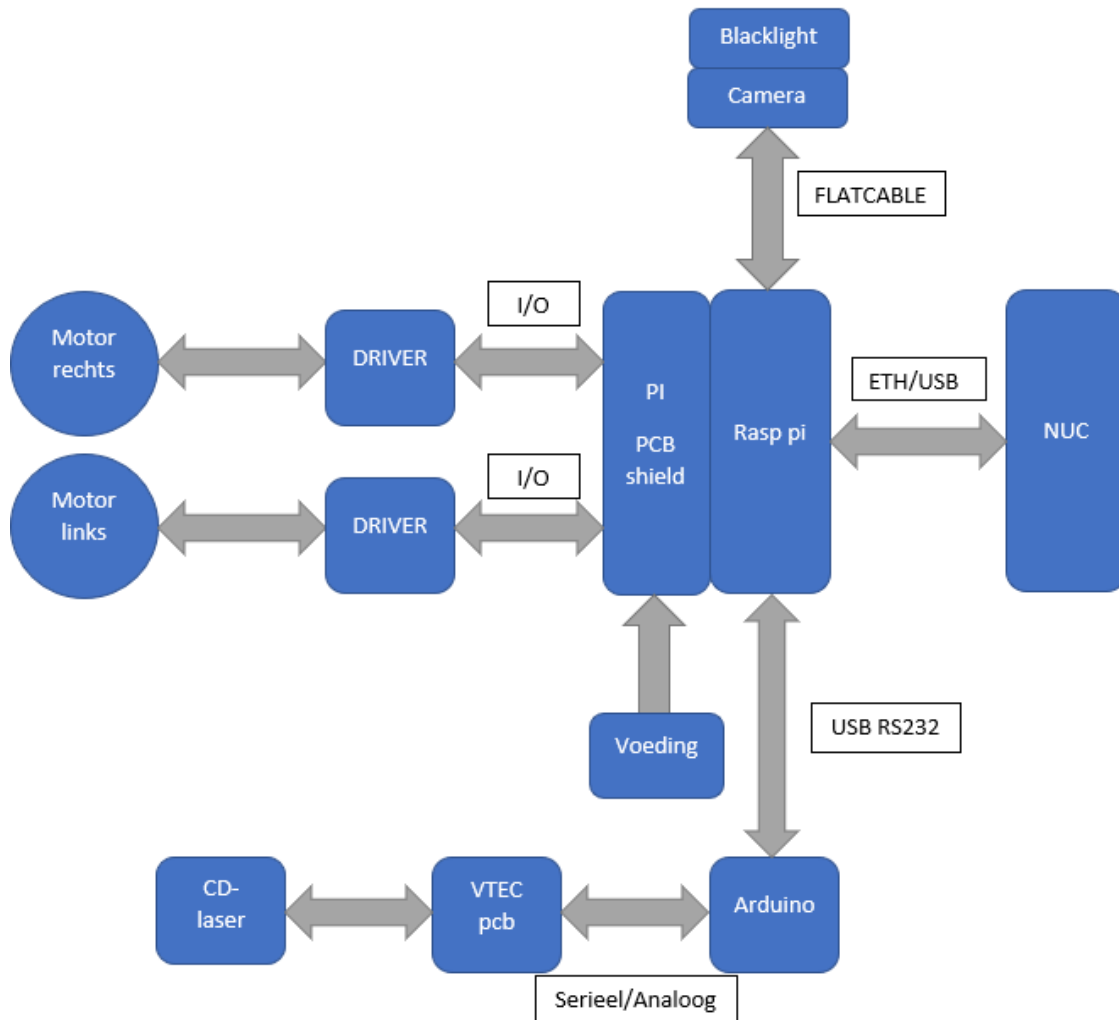
1.1.1 Programma van eisen

Voor het project zijn eisen opgesteld waar het eindresultaat aan dient te voldoen. Er is onderscheid gemaakt tussen functionele eisen en niet functionele eisen. Deze eisen zijn te vinden in het plan van eisen in bijlage A.

2. Analyse van de inspectierobot

Vorig jaar heeft een groep studenten onderzoek verricht naar het ontwerpen en realiseren van een robot die in staat is om een tankwand te inspecteren op olieresten. De groep studenten heeft in korte tijd de robot mechanisch volledig gerealiseerd. Op functionele werking is nog veel onduidelijkheid welke onderdelen van de robot wel en niet werken. Om dit duidelijk te krijgen is door een nieuwe groep studenten onderzocht welke onderdelen van de robot functioneel zijn.

Om de verschillende elektrische systemen in de robot in kaart te brengen is een blokdiagram opgesteld in figuur 1.



Figuur 1: blokdiagram elektrische systemen

De robot bestaat uit de volgende onderdelen:

- **Raspberry Pi** – De Raspberry Pi is een minicomputer en heeft het formaat van een pinpas. De minicomputer leest de informatie van de ruwheidssensor en de camera module. Daarnaast regelt de computer de aansturing van de motoren en de blacklight.
- **Raspberry shield** - De vorige groep studenten heeft een shield ontworpen om de kabels van andere onderdelen aan te sluiten. Daarnaast voorziet de shield alle componenten van spanning.

- **Ruwheidsdetectie** – Deze sensor detecteert de ruwheid van het onderliggende oppervlak. De CD-Laser meet de ruwheid en geeft de signalen door aan een PCB die geleverd is door V-Tec. De Arduino leest vervolgens de gegevens van de PCB en verstuurd deze serieel naar de Raspberry Pi via USB.
- **Aandrijving** – De robot wordt aangedreven door twee Elektronische Commutatie (EC) motoren. De motoren worden bestuurd door een driver die aangestuurd wordt door de Raspberry Pi.
- **Camera** – De robot is voorzien van een camera module. Deze maakt beelden van het oppervlak en stuurt deze naar de Raspberry Pi. Bij de cameralens zijn een viertal blacklight leds gemonteerd. Hierdoor kan de camera onzichtbare onzuiverheden detecteren.
- **NUC** – Een Next Unit of Computing (NUC) is een minicomputer. Dit is het brein van de robot. Omdat de Raspberry Pi niet genoeg rekenkracht bevat om de camerabeelden te verwerken is deze computer in het systeem opgenomen.

2.1 Raspberry Pi

De Raspberry Pi wordt in de huidige status gebruikt voor het aansturen van alle hardware in de robot. Alle sensoren en actuatoren, behalve de ruwheidsensor, zijn direct verbonden met de GPIO pinnen van de Raspberry. De ruwheidsensor wordt met een Arduino uitgelezen die vervolgens de data serieel met de Raspberry communiceert. Voor al deze processen is software aanwezig. De software is nog niet voldoende afgewerkt om dit in gebruik te kunnen nemen. Tijdens het testen van deelsystemen van de robot is gebleken dat een PWM-pin van de Raspberry niet werkt. De Raspberry beschikt over twee PWM pinnen. Deze worden beide gebruikt om beide motoren individueel aan te sturen.

2.2 Raspberry Pi shield PCB

De GPIO van de Raspberry Pi zijn aangesloten met een connector op een Printed Circuit Board (PCB). Deze PCB is ontworpen door de vorige groep studenten. Om de werking van de PCB te achterhalen is de aangeleverde documentatie onderzocht. Hierin zijn de volgende functionaliteiten gevonden:

- Spanningsvoorziening componenten
- Aansturing en uitlezen motoren
- Aansturing gyroscoop
- Blacklight aansturen en dimmen.

In de documentatie zijn de verbindingen van de PCB terug te vinden die benodigd zijn om de PCB de gewenste functionaliteit te geven. Deze zijn vergeleken met de actuele print in de PCB. Een overzicht hiervan is weergegeven in tabel 1

Tabel 1: Functies PCB

Functie	Actuele staat
Spanningsvoorziening componenten	De print is aangestoten op de voeding van de robot. Als de voeding wordt ingeschakeld gaan alleen de leds van de motor drivers aan. De Raspberry Pi blijft uitgeschakeld.
Aansturing en uitlezen motoren	De PCB is aangesloten aan de twee motordrivers van de motoren. In paragraaf 2.4 wordt de aansturing en de werking getest.
Aansturing gyroscoop	De gyroscoop sensor is niet aanwezig in de robot.
Blacklight aansturen en dimmen	De bedrading naar de blacklight module ontbreekt.

Naast de gedocumenteerde functionaliteiten is op de PCB een aansluiting van 12V aanwezig. Uit voorzorg is deze aansluiting gemeten met een multimeter. Uit de meting is gebleken dat op de uitgang een te hoge spanning van 16 Volt aanwezig is. De draden die zijn aangesloten op de PCB hebben een te grote diameter en kunnen eenvoudig uit de connector komen.

2.3 Aandrijving

De robot wordt aangedreven door twee EC-motoren van het merk Maxon. Deze zijn met tandwielen en een tandriem-overbrenging gekoppeld aan de rupsbanden. Elke motor is aangesloten op een motor driver. De driver zorgt voor de aansturing van de motor en ontvangt signalen van de Raspberry Pi. Omdat de software voor de Raspberry niet is voltooid, is er een oplossing bedacht om te testen of de aandrijving functioneert en of deze krachtig genoeg is om tegen een stalen wand op te rijden.

De documentatie van de vorige studenten is geraadpleegd om de aansluitingen van de motor, drivers en de Raspberry Pi te controleren. Bij deze controle is gebleken dat alle aansluitingen die benodigd zijn om de motor te laten bewegen aanwezig zijn. Vervolgens is een tijdelijke oplossing bedacht om de aandrijving te testen. Hiervoor is een python script geschreven dat kan worden uitgevoerd op de Raspberry Pi. Het script is terug te vinden in bijlage B. Om de motor aan te zetten zijn er drie signalen vereist voor elke motor driver:

1. **Motor aan** - dit is een spanning van 0 tot 5 volt waarbij 0 uit is en 5 volt aan.
2. **Richting** – Geeft de rotatie richting aan van de motor. Op 0 volt is deze links om, bij 5 volt is deze rechts om.
3. **Snelheid** – Met dit signaal wordt de snelheid van de motor geregeld. Het signaal bestaat uit een pulse-width modulation (PWM). Dit is een bloksignaal dat een spanning kan variëren van 0 tot 5 volt. Het bereik van het signaal is 0 tot 100 procent waarbij de spanning van 5 volt de 100 procent representeert.

De robot beschikt over twee motordrivers die worden aangestuurd door de Raspberry Pi. Om de pinnen waarop de drie signalen van elke driver zijn aangesloten te achterhalen is de documentatie geraadpleegd. Tabel 2 bevat de aansluitpinnen van de Raspberry Pi om de motoren aan te sturen. Deze zijn gebruikt om de motoren te besturen met het python script.

Tabel 2: aansluitpinnen Raspberry Pi

Functie	Nr	GPIO PIN	Naam	Beschrijving
DRIVER2	11	GPIO17	ENA2	Aanzetten motor 2
DRIVER2	12	GPIO18	PWM2	Snelheid regelen motor 2
DRIVER2	13	GPIO27	DIR2	Richting regelen motor 2
DRIVER1	15	GPIO22	ENA1	Aanzetten motor 1
DRIVER1	18	GPIO24	DIR1	Richting regelen motor 1
DRIVER1	33	GPIO13	PWM1	Snelheid regelen motor 1

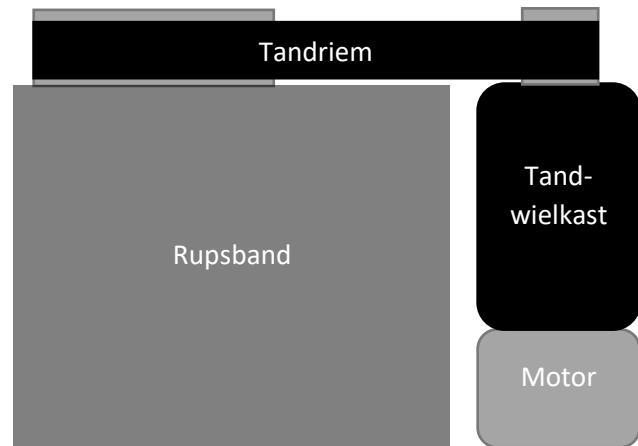
Tijdens het testen van de motoren met het pythonscript is vastgesteld dat motor 2 niet ging draaien. Om de oorzaak te vinden zijn alle pinnen uit tabel gemeten met een multimeter. Hieruit is gebleken dat de PWM2 op de Raspberry Pi defect is. Om een defecte verbinding uit te sluiten is een nieuwe Raspberry Pi aangesloten en getest met het script. Tijdens de tweede test met de nieuwe Pi draaiden beide motoren. Hiermee is aangetoond dat de PWM-pin van de robot defect is.

Als derde test is de robot op een verticaal oppervlak geplaatst om te controleren of de robot voldoende kracht heeft om zichzelf omhoog te rijden. De robot is enkel in staat om zichzelf op een

vaste positie te houden wanneer de motoren ingeschakeld zijn. Wanneer de motoren worden uitgeschakeld rolt de robot naar beneden.

De motoren zijn niet krachtig genoeg om de robot verticaal omhoog te laten rijden. Om de oorzaak te achterhalen is de aandrijving opnieuw doorgerekend. De resultaten zijn vervolgens vergeleken met de berekeningen die de vorige studentengroep heeft gemaakt. De berekeningen zijn in bijlage E terug te lezen. In figuur 2 zijn de overbrengingen in de robot weergegeven.

Uit de berekeningen blijkt dat de huidige motor en aandrijving een kracht kan overbrengen van 372,8 Newton per motor. Dit betekent dat beide motoren een gewicht van 74 kg verticaal moet kunnen verplaatsen. Dit is ruim voldoende om de robot van 15 kg aan te drijven.



Figuur 2: Overbrengingen motor

2.3.1 Motordrivers

Om de oorzaak van de ontbrekende kracht verder te onderzoeken is gekeken naar de firmware op de motordrivers. Per motor is een driver aanwezig die met Maxon software via een usb poort kan worden geprogrammeerd. Op de website van Maxon is Escon studio gedownload om de drivers uit te lezen. De drivers zijn ingelezen door Escon studio software zodat de parameters van de motordrivers kunnen worden gecontroleerd. De drivers bevatten instellingen van de mapping voor de snelheid. Deze komt via een PWM-signaal binnen op analoge ingang 1 van de driver. Wanneer de Raspberry Pi via GPIO 13 of 18 een spanningsignaal naar de driver stuurt kan aan de hand van deze parameter de gewenste snelheid worden ingesteld. Deze huidige instelling staat ingesteld dat de motor een stroom van maximaal 1.5 A krijgt bij een spanning de maximale spanning van 3.3 Volt. In de documentatie van de motor is terug te vinden dat de motor een nominale stroom vraagt van 3.21A. Deze parameter is niet juist ingesteld en moet worden verhoogd naar 3.21A om het maximale koppel van de motor te kunnen benutten.

2.3.2 Besturing via controller

Om de robot te kunnen laten rijden is bij het onderzoek naar de staat van de robot geen bestaande software geconstateerd. Uit voorgaande analyse blijken er meer aanpassingen nodig te zijn aan de software. Hierdoor is de keuze ontstaan te kiezen tussen een script of een controller om de robot te besturen. Door het korte tijdsbestek is overwogen de robot te laten rijden met een controller.

2.4 Camera

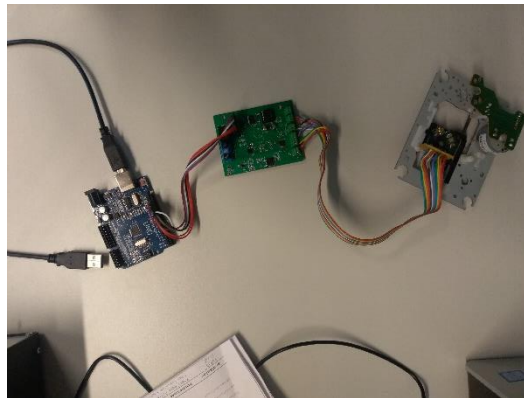
De camera is direct verbonden met de Raspberry Pi. De camera zal worden ingezet om foto's te maken voor inspectie. De software die voor de camera is bijgeleverd is niet bruikbaar vanwege een ontbrekende code. Om de camera toch te kunnen testen is ervoor gekozen een nieuwe code te schrijven waarmee de camera getest wordt op het geven van beeld. De testrapportage is te vinden in bijlage C.

2.4.1 Blacklight module

De blacklight is onderdeel van de camera. De blacklight wordt toegepast om bepaalde stoffen op te lichten waardoor deze zichtbaar zijn op de camerabeelden. In de voorgaande documentatie is onvoldoende informatie te vinden over de aansturing van de blacklight. De test die is uitgevoerd om de blacklightmodule te testen is te vinden in bijlage D.

2.5 Ruwheidsensor

De ruwheidsensor is aangeleverd door VTEC. De sensor bestaat uit een CD-lezer, een pcb en een Arduino. De Arduino bevat software die door VTEC met de sensor is meegeleverd. In figuur 3 is de opstelling met de drie onderdelen te zien.



Figuur 3: Opstelling ruwheidsensor

De sensor heeft de functie om ruwheid van een oppervlakte te meten. Door de afstand te meten tussen de laser van de cd-lezer en het oppervlak kan de sensor tot micrometers nauwkeurig de ruwheid in kaart brengen. Het PCB functioneert als een adapter tussen de Arduino en de cd-lezer. De cd-lezer krijgt via een externe spanningsbron -aangesloten op het PCB- spanning. De software op de Arduino kalibreert de cd-lezer om de afstand te kunnen lezen naar het oppervlak. Nadat de kalibratie is voltooid worden de gemeten waarden verstuurd naar de Raspberry Pi via USB. Uit het analyseren van de ruwheidsensor is gebleken dat de laser van de cd-lezer niet werkt waardoor de afstand niet gemeten kan worden. In bijlage D is de testrapportage terug te vinden. Omdat uit de test gebleken is dat de cd-lezer niet functioneert moet er naar een alternatief gezocht worden.

2.6 NUC

Een NUC (Next Unit of Computing) is een kleine computer met meer rekenkracht dan de Raspberry Pi. De NUC staat in verbinding met de Raspberry Pi die alle data verstuurd naar de NUC. Van de aanwezige software is geen documentatie aanwezig. Daarom is besloten om de software voor de NUC opnieuw te schrijven. Om zeker te zijn dat er niks mis is met de NUC is getest NUC correct opstart en of de poorten functioneren. Hierbij zijn geen bijzonderheden naar voren gekomen.

2.7 Conclusie analyse inspectierobot

Uit de analyse van de verschillende deelsystemen in de robot is gebleken dat een aantal onderdelen niet functioneert of ontbreekt. Daarnaast komen de werkelijke robotsystemen en documentatie niet altijd overeen. Om de robot te laten functioneren moeten in het ontwerp de volgende systemen worden verbeterd of veranderd:

- **Raspberry Pi** – De huidige Raspberry Pi kan worden ingeschakeld met behulp van een externe voeding. Om bekabeling zoveel mogelijk te beperken dient er een oplossing gezocht te worden om de Raspberry Pi intern van voeding te voorzien.
- **Raspberry Shield** – Volgens de documentatie van de robot heeft de shield verschillende functionaliteiten. Tijdens de analyse is geconstateerd dat één van de vier functies aanwezig is. De aanwezige spanningen komen niet overeen met de aanduiding op de print zelf. De shield dient voor het nieuwe ontwerp te worden aangepast of volledig opnieuw ontworpen te worden.
- **Aandrijving** – Tijdens de analyse zijn de motoren getest. De motoren functioneren op een vlakke ondergrond. Als de robot op een verticale wand geplaatst wordt zijn deze niet krachtig genoeg om de robot omhoog te bewegen. De huidige instellingen van de motoren zijn niet correct en dienen te worden ingesteld volgens de specificatie van de motoren.
- **Ruwheidssensor** – De ruwheidssensor is in de analyse gecontroleerd of deze functioneert. Uit de analyse van de sensor is gebleken dat de laser van de sensor defect is. In het nieuwe ontwerp dient de sensor te worden vervangen.
- **Software** – De huidige software voor de NUC is in de analysefase onderzocht of deze eenvoudig kan worden opgestart. Omdat de opstart procedure ontbreekt en de documentatie niet compleet is, is ervoor gekozen om de software opnieuw te schrijven.

Conclusie analyse

Uit de analyse is gebleken de voorgaande projectgroep de focus voornamelijk gelegd heeft op de hardware van de robot. De robot is elektrisch al voorbereid maar is nog niet af vanwege verschillende problemen die opgedoken zijn. De Raspberry Pi heeft een defecte PWM pin, de PCB shield mist belangrijke functionaliteit voor het aansturen van de blacklight of voor het voeden van randapparatuur. De motordrivers zijn onjuist ingesteld waardoor de robot niet is staat is om verticaal omhoog te rijden. De documentatie voor de software en de inbedrijfstelling ontbreekt waardoor het onmogelijk is om de robot te laten functioneren zoals is beschreven door de vorige studenten.

3. Ontwerp

In dit hoofdstuk wordt eerst ingegaan op de problemen die uit de analyse van het ontwerp naar voren zijn gekomen. Er wordt een keuze gemaakt hoe dit in het ontwerp opgelost wordt. Daarnaast zal het huidige ontwerp uitgebreid worden. De projectgroep heeft hiervoor doelstellingen geformuleerd. Deze doelstellingen zijn te vinden in bijlage A. Vanaf paragraaf 3.2 wordt de uitbreiding van het ontwerp toegelicht.

3.1 Aanpassing huidige ontwerp

Uit de analyse zijn de punten naar voren gekomen zoals beschreven in paragraaf 2.7. Ieder punt uit deze conclusie zal hieronder worden toegelicht.

3.1.1 Raspberry Pi & Raspberry Shield

De Raspberry Pi kan van spanning worden voorzien op de GPIO pin. De huidige Raspberry Pi shield heeft niet de mogelijkheid om de Raspberry Pi te voorzien van voeding. Daarom is ervoor gekozen om een nieuw PCB te ontwerpen om de Pi van voeding te voorzien. Het ontwerp van het PCB wordt in paragraaf 3.2.4 verder toegelicht.

3.1.2 Motordriver gegevens

In de analyse is gebleken dat de motordrivers niet juist zijn ingesteld. Om deze reden krijgen de motoren niet genoeg vermogen om verticaal omhoog te rijden. Om de drivers juist af te stellen is gekeken naar de motorspecificaties. Hierin staat het maximale vermogen van de motor vermeld en dient in de driver te worden ingesteld. In onderstaande tabel 3 zijn de belangrijkste gegevens weergegeven.

Tabel 3 belangrijke gegevens

Gegevens	Waarde
Nullasttoerental	6110 rpm
Nullaststroom	234 mA
Nominaal toerental	4860 rpm
Nominaal koppel (max. permanent koppel)	128 mNm
Nominaalstroom (max. permanente belastingsstroom)	3.21 A
Maximum koppel	1460 mNm
Startstroom	39.5 A
Koppelconstante	36.9 mNm/A
Toerentalconstante	259 rpm/V
Toerental-koppelconstante	4.26 rpm/mNm
Mechanische aanlooptijdconstante	8.07 ms

In de realisatiefase dienen deze instellingen te worden ingesteld in de drivers zodat het motorvermogen volledige benut kan worden. Wanneer de motoren meer vermogen krijgen kan de robot verticaal omhoog rijden.

3.1.3 Ruwheidssensor

In de analyse is gebleken dat de laser van de ruwheidssensor defect is. Om dit probleem op te lossen is er besloten dat de laser van de sensor wordt vervangen door hetzelfde type laser KSS-213C. Om de nieuwe laser op de juiste manier over te kunnen zetten zijn de kleuren van de bedrading op de laser gedocumenteerd. Deze staan in bijlage O.

3.1.4 Software NUC

De besturingseenheid zal een next unit of computing (NUC) zijn die zich in de robot bevindt, de NUC zal verantwoordelijk zijn voor de communicatie tussen de verschillende onderdelen van de robot en communicatie naar de client. Ook zal de NUC de informatie van de sensoren verwerken om te bepalen of het oppervlakte correct is schoongemaakt.

Voor de communicatie tussen de verschillende onderdelen van de robot zal de NUC ingesteld worden als ROS-master. De ROS-master is verantwoordelijk voor het bijhouden van publishers en subscribers op diverse onderwerpen en services, hierdoor is de ROS-master in staat nodes te helpen in het vinden van elkaar. Waarna communicatie tussen de nodes met elkaar peer to peer gaat zonder afhankelijkheid van de master. Een Ros node bevat een algoritme die bijdraagt aan de aansturing van het systeem, bijvoorbeeld de aansturing van de motoren. De ROS node zal subscriben op besturingsinput en de uitgevoerde actie publiceren op het ROS-netwerk.

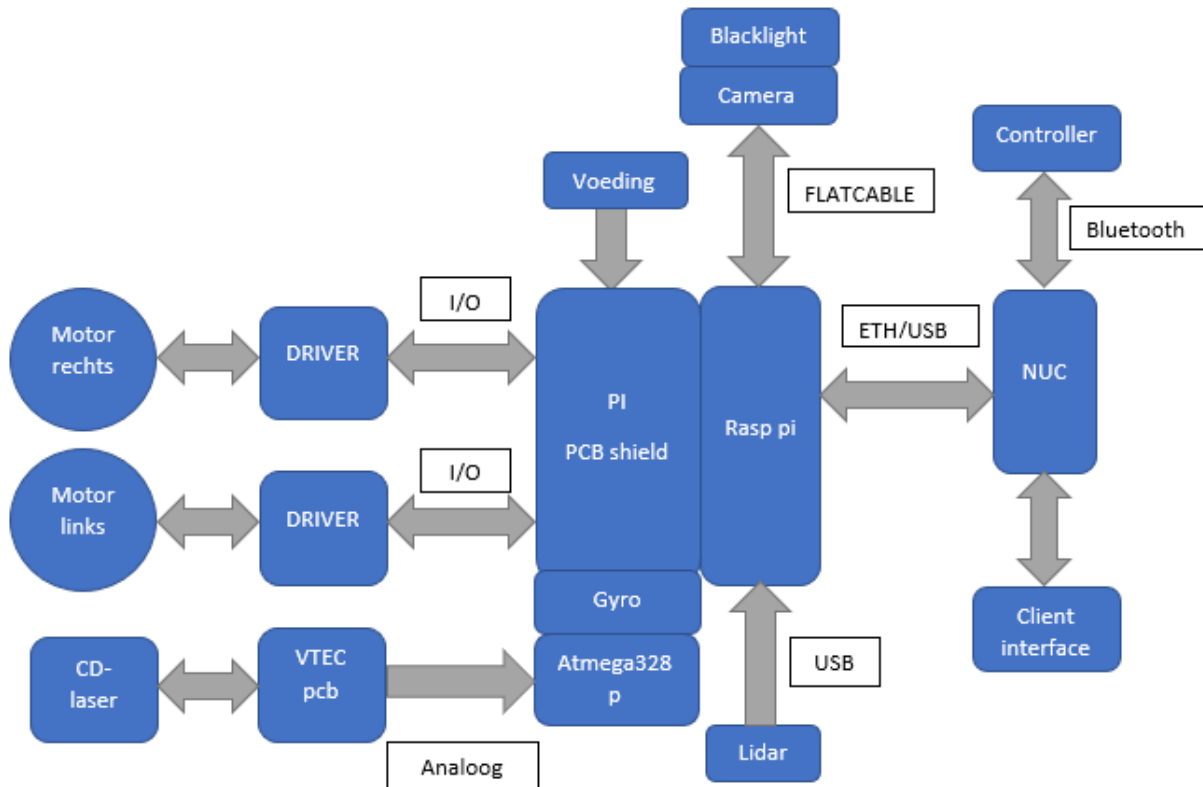
De besturingseenheid bevat een webservice voor het serveren van de client interface, hierdoor is het niet nodig dat de client interface op een client geïnstalleerd wordt. Update in de interface worden doorgevoerd op de server waarna de updates bij een nieuwe sessie opgehaald worden. Voor de communicatie tussen de client en de robot zal de besturingseenheid een RESTful api hosten waarnaar http-commando's verzonden worden, vervolgens wordt er een JSON-response teruggestuurd kan worden. De api zal verantwoordelijk zijn van het serveren van alle data die opgeslagen is de database, hierdoor is het mogelijk om historische data op te halen vanaf de robot.

Door het gebruik van een RESTful api is integratie met andere systemen zeer eenvoudig, door de aanwezigheid van een standaard protocol voor het ophalen van informatie. Voor de opslag van gegevens zal er gebruik gemaakt worden van een document store, wegens de hoge performance bij grote hoeveelheden ongestructureerde en niet gerelateerde data die de robot zal verwerken is deze database vorm het meest geschikt.

Voor het bepalen of het oppervlakte correct is schoon gemaakt zal er een supervised regression machine learning algoritme gebruikt worden, het algoritme is het meest efficiënt omdat de input en output variabelen van elkaar afhankelijk zijn. Ook is het mogelijk om een trainingsdataset te verzamelen waarmee het algoritme getraind kan worden. De eerste invoer voor het algoritme zal de ruwe data van de ruwheidssensor zijn. De data van de blacklight sensor zal geanalyseerd worden aan de hand van OpenCV, het OpenCV algoritme zal olievlekken detecteren die vervolgens als invoer ingevoerd worden in het algoritme.

3.2 Uitbreiding van het ontwerp

Om aan de gestelde eisen in bijlage A te voldoen, is het blokschema uit hoofdstuk 2 uitgebreid met sensoren en een client interface. De uitbreidingen zijn toegevoegd in het blokschema en is afgebeeld in figuur 4.



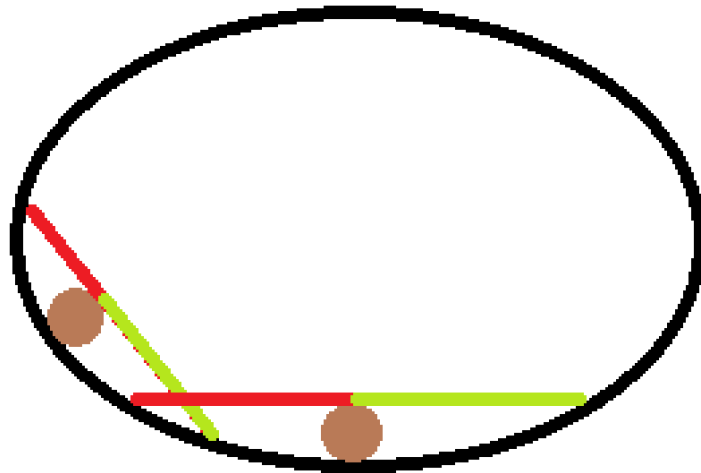
Figuur 4: Uitgebreid blokschema systemen

3.2.1 Uitbreiding van sensoren

Om de doelstellingen van het project te bereiken, moeten in het ontwerp extra sensoren worden toegepast. In het ontwerp is een lidarsensor en een gyrosensor toegevoegd om de positie van de robot in de tank te bepalen. Er is een combinatie van sensoren gekozen om de positie van de robot te bepalen nauwkeuriger te kunnen bepalen en de bepaling ook betrouwbaarder wordt.

3.2.1.1 Lidar

De lidar sensor wordt toegevoegd om eventuele obstakels te kunnen detecteren. De obstakels zouden als referentiepunt voor de robot kunnen dienen. Wanneer dit obstakel altijd zichtbaar is de positionering van de robot ook zichtbaar. In figuur 5 is een voorbeeld te zien van de robot met de lidar sensor. De groene stip is de robot. De lidar sensor staat hierboven op en draait 360 graden rond. De rode en groene lijn stelt de laser van de lidar voor. De voorkant van de robot is aangegeven met de rode lijn. De groene lijn is de achterkant. In elke niet perfecte bol zal de voorkant en achterkant van de robot een bepaalde afstand tot de tankwand hebben. In het voorbeeld van figuur 5 werkt dit in elk kwadrant van de tank. Zoals weergegeven de robot in het midden onderaan de tank heeft gelijke afstand vanaf de voorkant naar de tank wand, als de achterkant tot aan de tankt wand. Dit geldt ook als de robot in het midden aan de boven kant van de tank staat. Door een combinatie van sensoren wordt dit probleem opgelost. Dit wordt verder besproken in de opvolgende hoofdstukken. De lidar sensor is voor deze toepassing is geselecteerd op de prijs en integratie met ROS en zal met de Raspberry pi verbonden worden.



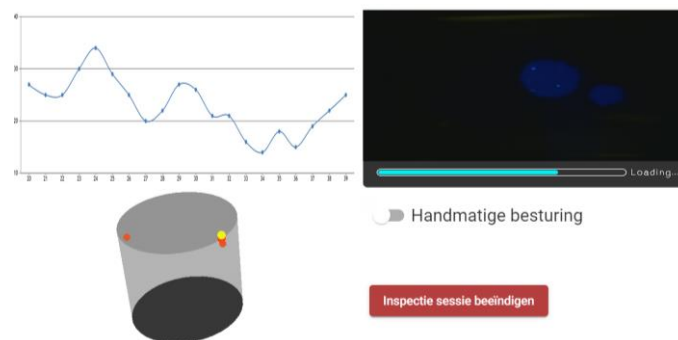
Figuur 5: Situatieschets lidar

3.2.1.2 MinIMU-9

De MinIMU-9 is een module waar een Gyro, Accelerometer en kompas in geïntegreerd zijn. De leverancier van de unit heeft diverse libraries beschikbaar waarmee het uitlezen van de unit vereenvoudigd is. Ook zijn de beschikbare libraries compatible met de Arduino die al aanwezig is in de robot. De gyromodule bevat een accelero-sensor, een magneto-sensor en een gyrosensor. De toepassing van deze module is te vinden in paragraaf 3.2.3. Met de gyro sensor kan aan de hand van de zwaartekracht bepaald worden in welk kwadrant de robot zich in de tank bevindt. De accelerometer wordt toegevoegd om te kunnen bepalen wanneer de robot slip heeft. Als de robot valt meet de accelerometer een bepaalde versnelling in een bepaalde richting. Door dan de data van de lidar en gyrosensor te combineren kan de locatie van de robot opnieuw worden bepaald.

3.2.2 Client Interface

De client interface is een single-page application(SPA), dit verhoogt de plug and play compatibility. Door het gebruik van een SPA zal er geen installatie nodig zijn op de client, ook is het doorvoeren van updates in de interface eenvoudig. De visualisatie van de gegevens wordt op verschillende manieren binnen de SPA weergegeven. Als eerste zal er een videostream zijn, de stream zal een vertoning zijn van het ruwe image vanuit de sensor (rechtsboven Figuur 6). Daarnaast is er een real time grafiek die over het verloop van tijd de ruwe data zal tonen van de ruwheid van de oppervlakte vanuit de ruwheidssensor (linksboven Figuur 6). Door het tonen van de ruwe data van de blacklight sensor en de ruwheidssensor is de eindgebruiker in staat zijn persoonlijke analyse op de gegevens uit te voeren. Naast de ruwe gegevens zal de interface ook een canvas bevatten met een 3d model van de olietank (linksonder Figuur6), hierin zal de huidige positie van de robot getoond worden op basis van de data van diverse sensoren uit de robot. Naast de huidige positie van de robot zal er in het 3d model ook de locatie getoond worden van de plekken die door het algoritme gelabeld zijn als niet goed gereinigd. Aan de hand van het 3d model kan de reinigingsrobot de plekken reinigingen die voorheen niet goed gereinigd zijn, hierdoor hoeft de tank niet nogmaals volledig doorlopen te worden.



Figuur 6: Client Interface

3.2.2.1 Communicatie interface en robotbesturing

De interface voor de client wordt een web client die communiceert met een RESTful API en de ROS-master op de besturingseenheid. Door gebruik te maken van een web client is er geen aparte installatie of configuratie nodig op de verschillende clients, dit voorkomt afhankelijkheid van de correcte setup van een client. De client wordt een single-page application(SPA) die tijdens de inspectie HTTP requests zal versturen naar de API waarna de API een JSON response zal terug sturen. De SPA zal vervolgens de respons verwerken en gegevens aan de gebruiker tonen.

De client interface is tevens verantwoordelijk voor het opvangen en verwerken van de handmatige besturingsinput, die ingevoerd wordt aan de hand van een xbox-360 controller. De client zal vervolgens deze waarden versturen naar de besturingseenheid, waarna de motoren op de correcte manier aangestuurd worden.

3.2.3 Positionering van de robot

Voor het bepalen van de positie in de olietank worden verschillende sensoren gecombineerd om een accurate positie te berekenen. Een gyrosensor is aanwezig voor het bepalen van de oriëntatie van de robot, hiermee kan vastgesteld worden welke richting de robot op rijdt. Daarnaast is er een accelerometer aanwezig in de gyro-module, hiermee kunnen plotselinge veranderingen in de positie waargenomen worden. Deze veranderingen kunnen optreden door het wegslijpen van de robot in de olietank.

Voor het bepalen van de hoogte locatie van de robot wordt een lidar sensor gebruikt, de sensor meet aan de hand van laser stralen de afstand van de robot tot een object. In de situatie van een olietank meet deze sensor de afstand van de robot tot het plafond of de grond.

De berekende positie wordt vervolgens gepubliceerd zodat de client de huidige locatie kan weergeven en het algoritme voor het autonoom rijden weet welke vervolgactie genomen moet worden.

3.2.4 Besturing via controller

Ros ondersteunt een groot aantal verschillende controllers. Zowat elke controller die met een pc te verbinden is, kan toegevoegd worden als rosnode. De robot zal in industriële omgeving opereren dus vallen minder kwalitatieve controllers af. De controller moet tegen een stoot kunnen. Daarnaast kan er gekozen worden tussen een controller met draad of draadloze controllers. Controllers met een kabel hebben geen batterijen nodig waardoor een controller niet leeg kan raken. Daarnaast kan er een breuk ontstaan in de bedrading. De controller wordt met de clean robot toepassing regelmatig meegenomen en opgerold in een tas gestopt worden. Om de gebruiksvriendelijkheid te waarborgen is er gekozen voor een draadloze controller. De draadloze controllers staan in twee soorten: een controller met een eigen usb dongle of een controller die via bluetooth direct verbinding maakt met een pc. Omdat usb dongels klein zijn, kwijt kunnen raken of vergeten kunnen worden, is gekozen voor een controller die via bluetooth direct met een pc verbindt. Met dit filter is gezocht op internet en hieruit bleek de Xbox One S controller uitermate geschikt te zijn. De controller is draadloos te verbinden met een pc over bluetooth, ROS ondersteunt de controller en de controller is stevig en robuust.

3.2.5 Nieuwe Raspberry Pi shield pcb-ontwerp

In het vorige hoofdstuk is de werking van de Raspberry Pi shield getest. Hieruit is gebleken dat de beschreven functionaliteiten deels of niet aanwezig waren. Om deze reden is ervoor gekozen om een nieuwe Raspberry Pi shield te ontwerpen. De huidige functionaliteiten uit de analyse worden uitgebreid met aansluitingen voor de nieuwe sensoren die zijn beschreven in paragraaf 3.2.1. Het ontwerp voor de shield dienen de volgende functionaliteiten te hebben:

- Spanningsvoorziening van naar de Raspberry Pi, gyrosensor, lidar sensor en motor drivers
- Aansluitingen voor de motor drivers naar de GPIO-pins van de Raspberry Pi
- Aansluitingen naar de Raspberry Pi voor het uitlezen van de gyrosensor en lidar sensor
- Aansluitingen naar de Raspberry Pi voor het aansturen van de blacklight module

3.2.5.1 Voeding componenten

Om een helder te krijgen welke spanning elk component op de shield nodig heeft is in tabel 3 een overzicht gemaakt van alle voedingsspanningen.

Tabel 4: Overzicht componenten

Component	Voedingsspanning	Stroomverbruik
Lidar YDLIDAR X4	5 VDC	480 mA
Raspberry Pi 3B+	5 VDC	2 A
Gyrosensor	3,3 VDC	6 mA
Motor drivers	24 VDC	7,25 A

De voeding in de robot heeft een gelijkspanning van 24V. Op deze spanning kunnen de motordrivers direct op worden aangesloten. Voor de voeding van de Lidar en de Raspberry Pi is ervoor gekozen om

een dc/dc-converter type LM2678 op de PCB te plaatsen. Het voordeel van deze omvormer is dat deze efficiënt de spanning om kan zetten van 24 naar 5 volt.

De GPIO van de Raspberry Pi heeft een 3,3 Volt aansluiting. Op deze pin kan de voeding van de gyrosensor worden aangesloten.

3.2.5.2 Communicatie motoren en sensoren

De gyrosensor, lidar en motordrivers worden aangesloten op de GPIO pinnen van de Raspberry Pi. De Raspberry wordt met de GPIO pinnen op de shield geplaatst zodat de pinnen naar aansluitklemmen kunnen worden gebracht.

De gyrosensor wordt op de PCB met headerpinnen geplaatst. De sensor maakt gebruik van het I2C protocol waarmee de sensor kan worden uitgelezen door de Raspberry Pi. De lidar sensor wordt aangesloten op de seriële aansluiting van de Raspberry Pi op een aansluitklem.

De motordrivers worden aangesloten op de GPIO van de Raspberry Pi. De drivers hebben de volgende aansturing nodig om de motoren te laten draaien:

- Motor aan
- Richting
- Snelheid

Alle componenten worden aangesloten op de Raspberry Pi aan de hand van onderstaande tabel:

Tabel 5: Pinconfiguratie op Raspberry Pi

Pin NR	Beschrijving	Aangesloten op	Pin NR	Beschrijving	Aangesloten op
1	3.3VDC	PWR ADC MCP3008	2	5VDC	
3	GPIO 2	GYRO SDA I2C	4	5VDC	
5	GPIO 3	GYRO SCL I2C	6	GND	
7	GPIO 4	BLACKLIGHT ON TRANSISTOR	8	GPIO 14 Tx	ARDUINO Rx
9	GND	GYRO GND	10	GPIO 15 Rx	ARDUINO Tx
11	GPIO 17	Motor 2 Enable	12	GPIO 18	
13	GPIO 27	Motor 2 Direction	14	GND	
15	GPIO 22	Motor 1 Enable	16	GPIO 23	Motor 2 Speed PWM
17	3.3VDC	GYRO PWR VIN	18	GPIO 24	Motor 1 Direction
19	GPIO 10 SPI	ADC DIN	20	GND	
21	GPIO 9 SPI	ADC DOUT	22	GPIO 25	
23	GPIO 11 SPI	ADC CLK	24	GPIO 8 SPI	ADC CS
25	GND		26	GPIO 7	
27	SDA0 I2C		28	SCL0 I2C	
29	GPIO 5		30	GND	
31	GPIO 6		32	GPIO 12	
33	GPIO 13	GP IO Pin motor 1	34	GND	
35	GPIO 19	GP IO Pin motor 2	36	GPIO 16	
37	GPIO 26	Motor 1 Speed PWM	38	GPIO 20	
39	GND	ADC GND	40	GPIO 21	

3.2.5.3 Ruwheidssensor

De huidige ruwheidssensor is aangesloten op een Arduino Uno en neemt te veel ruimte in beslag in de robot. Daarom is ervoor gekozen om de microprocessor van de Arduino, de Atmega328P op de pcb te plaatsen zodat de Arduino Uno uit de robot kan worden gelaten. De ruwheidssensor wordt direct om de PCB shield aangesloten die naar de microprocessor gaat. De seriële pinnen van de Atmega worden direct aangesloten op de Raspberry Pi. Omdat de seriële spanning van de Atmega hoger is dan de Raspberry is het noodzakelijk om een spanningsdeler te maken zodat de Raspberry niet defect kan raken.

3.3 Verificatie van het ontwerp

Ter verificatie is het ontwerp gecontroleerd aan de eisen die zijn opgesteld in het plan van aanpak. In tabel 6 wordt elke must-have eis gecontroleerd aan het ontwerp.

Tabel 6 verificatie eisen

Eis	Prioriteit	Verificatie
De oppervlakte ruwheid kan worden gemeten ISO 8501-1 norm.	Must have	Het ontwerp bevat een ruwheidssensor geleverd door VTEC. De normering van de ruwheidmeting kan pas worden bepaald in de realisatiefase.
De kwaliteit van de oppervlakte kan worden beoordeeld d.m.v. camerabeeld en blacklight volgens de ISO 8501-1 norm.	Must have	De kwaliteit van het oppervlaktebeeld kan pas worden vastgesteld in de realisatiefase.
De data van de ruwheidssensor en de camerabeelden kunnen worden uitgelezen een extern systeem d.m.v. een UDP-verbinding	Must have	De camerabeelden worden d.m.v. een ethernet kabel doorgestuurd naar de NUC waar ze kunnen worden verwerkt.
De inspectierobot kan op eigen kracht verticaal omhoog rijden met een snelheid van 0.15 m/s	Must have	Het nieuwe ontwerp bevat verbeterde instellingen voor de motordrivers. Hierdoor is de motor in staat om verticaal omhoog te rijden met 0.15 m/s
De robot beschikt over een LIDAR-sensor voor obstakeldetectie	Could have	Het nieuwe ontwerp bevat een Lidar sensor
De robot beschikt over een camera voor het inspecteren van de oppervlakte	Must have	De camera is aanwezig in het ontwerp.
De robot beschikt over een blacklight voor oppervlakte controle	Must have	Bij de camera module is een blacklight module geplaatst.
De robot beschikt over een ruwheidssensor	Must have	Het ontwerp bevat een ruwheidssensor geleverd door VTEC.
De robot dient door een mangat te passen van 60 cm.	Must have	De robot heeft een breedte van 59 cm
De server is extern bereikbaar om de data op te halen	Must have	De NUC verzamelt alle sensordata van de robot en

		kan op afstand worden uitgelezen.
--	--	-----------------------------------

4. Realisatie

In dit hoofdstuk wordt beschreven hoe de verschillende systemen gerealiseerd zijn. Per systeem wordt beschreven welke stappen genomen zijn tijdens het realiseren en de werking van het gefinaliseerde systeem. Het hoofdstuk is opgedeeld in aanpassingen en uitbreidingen van de robot.

4.1 Gerealiseerde aanpassingen

In deze paragraaf worden de aanpassingen van verschillende componenten beschreven. De aanpassingen zijn gemaakt op basis van het ontwerp dat is beschreven in het vorige hoofdstuk.

4.1.1 Ruwheidsensor

De ruwheidsensor is geleverd door VTEC. Bij de ruwheidsensor is ook Arduino code meegeleverd om de sensor uit te lezen. Deze code is te zien in bijlage P. De code is aangepast zodat alleen de afstandswaarde gestuurd wordt. Zoals in de analyse beschreven, is er een nieuwe laser besteld. Deze laser is aangesloten op de PCB. Hoe deze is aangesloten staat in bijlage O. De ruwheidsensor stuurde volgens de originele code de data serieel via USB, dit is aangepast zodat dit via de pins gaat. Ondanks deze aanpassingen werkt de ruwheidsensor nog steeds niet naar behoren. In de aanbevelingen wordt een mogelijke oplossing gegeven.

4.1.2 Motordriver instellingen

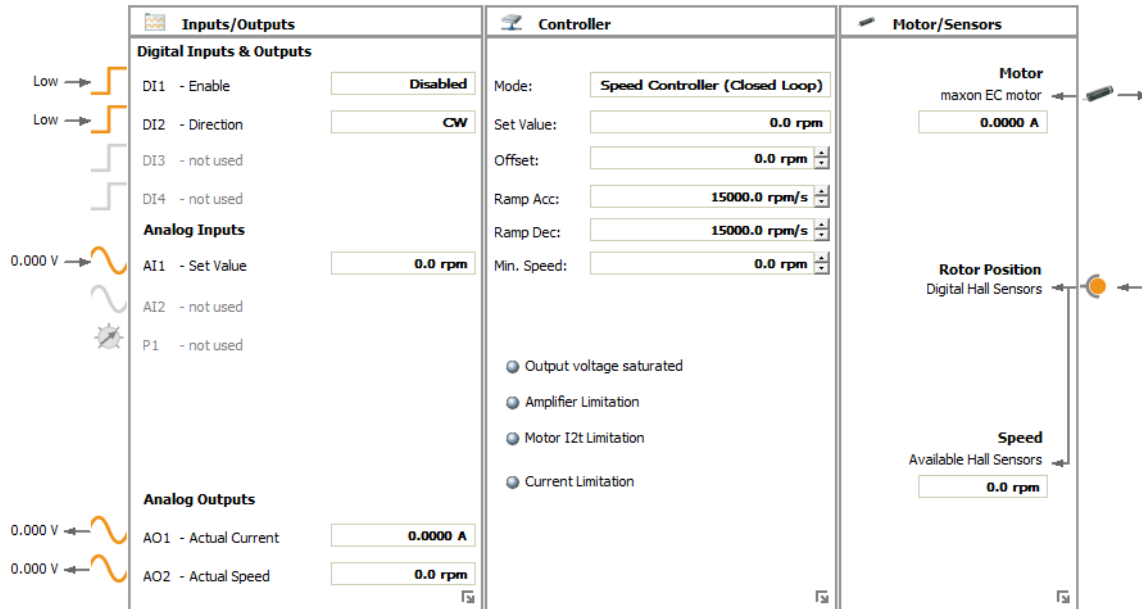
De huidige motordrivers zijn onjuist ingesteld. In het vorige hoofdstuk zijn de correcte motogegevens opgesteld zodat deze kunnen worden ingesteld in de drivers. Om de Maxon motordrivers te kunnen programmeren is de software Escon studio van de website van Maxon gebruikt. Door een laptop met een USB met de driver te verbinden maakt de software automatisch verbinding. Vervolgens kunnen de motorkarakteristieken volgens het ontwerp worden ingesteld zoals in figuur 7 is weergegeven.

The screenshot shows the 'Motor' configuration window in the Escon software. On the left is a navigation tree with categories: Motor/Sensors, Controller, and Inputs/Outputs. Under 'Motor/Sensors', 'Motor' is selected. The main area is titled 'Motor' and contains the following settings:

- Choose motor type: maxon EC motor
- Enter motor characteristics (consult maxon catalog for motor data):
 - Speed Constant: 259.0 rpm/V
 - Thermal Time Constant Winding: 29.6 s
 - Number of Pole Pairs: 8
- Enter system data:
 - Max. Permissible Speed: 4500.0 rpm
 - Nominal Current: 3.2100 A
 - Max. Output Current Limit: 3.2100 A

Figuur 7 driver instellingen 1

Daarnaast is de motor regeling aangepast naar closed loop. Dit betekent dat de motordriver aan de hand van het toerental de motor gaat regelen. Het toerental wordt gemeten door de aanwezige hall effect sensoren in de motor. Door de driver in een closed loop configuratie in te stellen worden piekstromen bij het inschakelen van de motoren beperkt en wordt de voeding niet overbelast. Het maximale toerental van de motoren is ingesteld op 4500 RPM. In figuur 8 is een overzicht van alle instellingen van de motordriver weergegeven.



Figuur 8 driver instellingen 2

4.2 Gerealiseerde uitbreidingen

Om de robot operationeel te krijgen is de robot uitgebreid met nieuwe software en sensoren. In deze paragraaf wordt de realisatie van deze onderdelen beschreven.

4.2.1 Robot Operating System

Het Robot Operating System, afgekort ROS is een flexibel framework voor het schrijven van robotsoftware. Het is een verzameling tools, bibliotheken en conventies die zijn gericht op het vereenvoudigen van de taak om complex en robuust robotgedrag te creëren op een breed scala aan robotplatforms. Omdat het maken van echt robuuste, algemene robotsoftware moeilijk is.

Vanuit het perspectief van de robot variëren problemen die voor de mens triviaal lijken vaak enorm tussen de verschillende taken en omgevingen. Het omgaan met deze variaties is zo moeilijk dat geen enkele persoon, laboratorium of instelling het alleen kan hopen te doen.

ROS kan gemakkelijk op verschillende apparaten worden geïnstalleerd. Ook bestaat er al veel software die op het internet te vinden is. In de volgende hoofdstukken worden de nodes besproken die met het ros operating system zijn uitgevoerd.

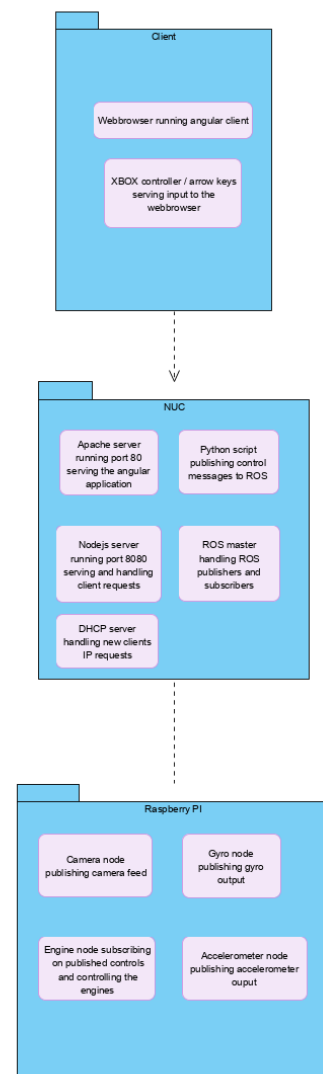
Het ROS-besturingssysteem draait op het Ubuntu besturingssysteem van Linux. Om ROS te kunnen gebruiken is Ubuntu geïnstalleerd op de NUC. De stappen voor het installeren van Ubuntu en ros zijn terug te vinden in bijlage H en I.

4.2.1.1 Nuc

De next unit of computing afgekort NUC dient als de kern van het verwerken van de robot gegevens en als toegangspunt voor de gebruiker tot de robot. Om het gebruik van de robot zo gebruiksvriendelijk mogelijk te maken is er een DHCP-server geïnstalleerd op de NUC. De DHCP-server zal de client pc van de gebruiker een IP-adres toewijzen, zodra de client verbonden wordt met de NUC aan de hand van de ethernet kabel. De client bevindt zich meteen in hetzelfde netwerk als de robot, waardoor communicatie mogelijk is. De NUC is ingesteld op IP-adres 192.168.1.1, de gebruiker kan dit adres in de webbrowser op de client gebruiken. De client zal een HTTP request versturen naar de NUC waarop de standaard poort voor http-verkeer (poort 80) het verzoek zal binnenkomen. Op de NUC draait een service listener op poort 80 die het verzoek van de client afhandelt. De listener service die gebruikt wordt is apache 2.4.41, apache zal bij een root verzoek de index.html in de /var/www map naar de client versturen. In de index.html staan de benodigde bestanden voor de client interface vermeld die door de client opgevraagd kunnen worden bij de Apache service. De installatie van de dhcp server is te vinden in bijlage K.

De bestanden in de /var/www map zijn het resultaat van de Angular compiler, na het compilen van het Angular project die de functionaliteit en uiterlijk van de gebruikersinterface definiëren. De gebruikersinterface zal de gebruiker in staat stellen om besturing commando's naar de robot te sturen. Aan de hand van de Gamepad API wordt de invoer van de controller die verbonden is met de client afgevangen, de invoer wordt in de controllerService verwerkt. Naast het gebruik van een controller is het mogelijk om gebruik te maken van de cursortoetsen. De invoer hiervan zal in combinatie met de invoer van de snelheid slider verwerkt worden in de controllerService. Voor het afvangen van de invoer zijn er key-up en key-down events gebruikt die de state updaten van de overeenkomende cursortoets. In de controllerService wordt de afgevangen invoer van de gebruiker verwerkt en omgezet naar JSON die verzonden wordt naar de NUC.

Op de NUC draait een Node.js service die luistert naar poort 8080, het POST verzoek van de client komt hierop binnen. Binnen de Node applicatie wordt het verzoek via de routing doorgezet naar de EngineCommandController, hierin wordt het verzoek met de JSON-string in de body verwerkt. De inhoud van de JSON-string wordt omgezet naar een ROS joy.msg bestaande uit een header met het tijdstip van de invoer, een float32 array met de waarden van axes die de richting en de snelheid van de robot bepalen en een int32 array met de knoppen die ingedrukt zijn. Deze arrays worden als argumenten meegegeven aan het python publisher script tijdens het starten van het script. Het script maakt verbinding met ROS en publiceert de meegegeven invoer op rostopic /joy. Op de Raspberry PI draait een script dat deze invoer afvangt en de motoren mee aanstuurt. In Figuur 9 staat een overzicht van de diverse componenten en bijbehorende applicaties.



Figuur 9 opzet structuur

4.2.1.2 Raspberry Pi

De Raspberry Pi bevat een besturingssysteem-image waarop ros al is voor geïnstalleerd. De software-image is afkomstig van Ubiquityrobotics. Tijdens het realisatieproces is gebleken dat de software van ubiquity de minste problemen opleverde en is daarom gebruikt als opzet voor de Raspberry Pi. De ubiquity software heeft voor gebruik enige aanpassingen nodig. De ubiquity software start een aantal processen op die bedoeld zijn voor de ubiquity robot. Deze zullen uitgeschakeld moeten worden.

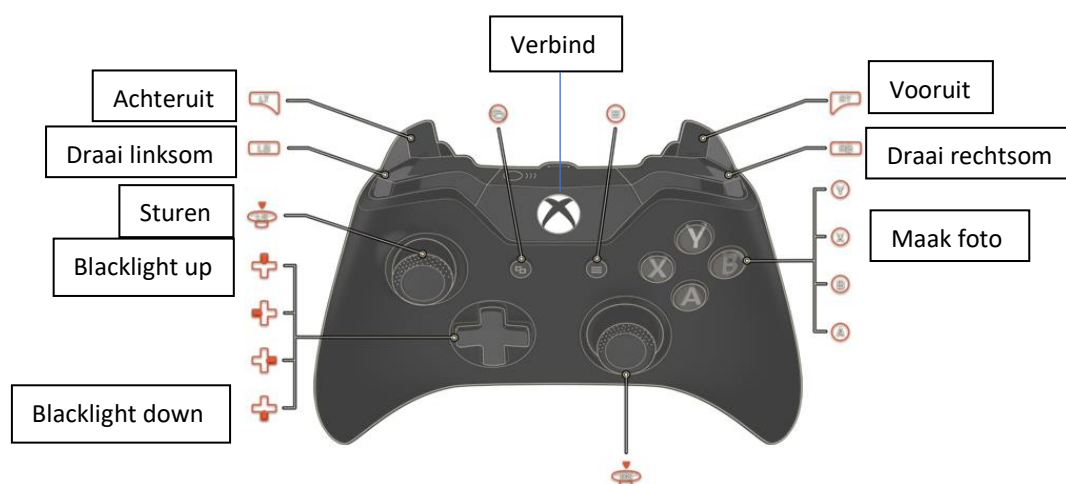
De Raspberry Pi wordt voornamelijk gebruikt als I/O bord en is dus de link tussen alle sensoren en de software. Voor alle sensoren wordt een ros node aangemaakt die verder in dit hoofdstuk individueel worden besproken. De Raspberry Pi staat met een bekabelde verbinding in contact met de NUC, waarop de rosmaster draait. Bij het opstarten van de nodes op de Raspberry Pi wordt er geen roscore opgestart maar wordt er geprobeerd verbinding te maken met de roscore op de nuc.

4.2.1.3 Xbox controller

De controller waarmee de robot bestuurd kan worden is een Xbox One S controller. De keuze voor gebruik te maken van een Xbox controller is gebaseerd op de robuustheid en draadloos gemak van de controller. Het is belangrijk een Xbox One S controller te gebruiken omdat dit model via bluetooth verbinding kan maken met een besturingssysteem zoals Windows of Linux. In bijlage J wordt in stappen weergegeven hoe de controller met een Linux systeem te verbinden is. De bijgeleverde controller is al verbonden met de NUC en kan tot stand gebracht worden door op de verbind knop te drukken. Met dezelfde knop kan de controller ook uitgeschakeld worden. Dit kan door de verbind knop ongeveer 10 seconden ingedrukt te houden. Hoe de verbinding tussen de controller en de nuc tot stand zijn gebracht is terug te vinden in bijlage J.

Wanneer het script op de NUC opgestart wordt, wordt er een joy node opgestart. De joy node verwacht een controller en vangt de waarden van de ingedrukte knoppen op en publiceert deze op het ros topic /joy.

In figuur 10 is de controller lay-out te zien. In aankomende paragrafen worden de individuele functies van de knoppen besproken.



Figuur 10 Xbox controller

4.2.1.4 Motor besturings-node

De inspectierobot bevat twee motoren die de rupsbanden aandrijven. Deze twee motoren kunnen in snelheid geregeld worden doormiddel van een PWM-sigitaal. De motoren gaan pas daadwerkelijk draaien als een enable bit wordt meegestuurd en kan ook van draairichting geregeld worden met een direction bit.

Op de Raspberry Pi is een node voor de motoren aangemaakt waarmee de robot bestuurbaar wordt. De node wil subscriben op een /joy node. Dit is gedaan zodat alle controllers die de ros joy package ondersteunen gebruikt kan worden om met de robot te rijden. Hierbij staat het ook vast hoe de robot bestuurd kan worden door eventueel een eigen script te maken.

De robot kan met de drie knoppen: vooruit, achteruit en sturen, bestuurd worden. Deze drie knoppen zijn allemaal analoge knoppen. Elke knop geeft een waarde tussen -1 en 1. De vooruit en achteruit knop geven -1 wanneer deze niet worden ingedrukt. Worden de twee knoppen volledig ingedrukt resulteert dit in 1. De stuur knop staat in het midden op 0. Wordt de knop naar volledig naar links gedrukt geeft dit een waarde -1. Volledig naar rechts wordt dit 1. De verhouding rekent het script voor de motoren om naar een waarde tussen 0 en 100 waarmee het pwm signaal naar de motoren wordt geregeld. Het complete verhaal is als package op de Raspberry Pi in de ros environment geïnstalleerd.

4.2.1.5 Lidar- node

De lidar node bestaat uit twee delen. Het deel op de Raspberry Pi en het deel op de NUC. Het deel op de Raspberry Pi wordt gebruikt om de lidar te starten en de data hiervan uit te lezen en te publiceren op de rosmaster. Het deel op de NUC subscribe op dit topic en leest de data uit en maakt deze visueel zichtbaar. De nodes van de Raspberry Pi en de NUC zijn toegevoegd in de launchfiles waarmee het volledige systeem van de Raspberry Pi en NUC opgestart kan worden.

De resolutie van de lidar is 0.5 graden. Dit betekent dat er in 360 graden rond de lidar 720 (360/0.5) meetpunten zijn. Al deze meetpunten zijn terug te vinden in de data die de Raspberry Pi op de node /lidar publiceert. Elk meetpunt is terug te vinden op de bijhorende locatie in de array. De data die dus wordt gepubliceerd is een array met 720 vakjes.

De installatie van de lidar is samen met de overige nodes samengevoegd als package. De installatie van dit package is terug te vinden in bijlage L.

4.2.1.6 Blacklight-node

De blacklights bevinden zich rondom de camera. De in totaal 4 blacklight leds zijn in felheid te regelen. De blacklights zijn als package toegevoegd op de Raspberry Pi in de ros environment. Bij het starten van de launchfile op de Raspberry Pi wordt de node voor de blacklights gelauncht. De node subscribe op de het topic /joy. Met de knoppen blacklight up en down kan de intensiteit van de blacklight geregeld worden. Er is bewust gekozen om de intensiteit regelbaar te maken omdat vloeistoffen verschillend het licht kunnen reflecteren en hiermee de camera verblinden. Bij elke druk op de knop up of down zal de intensiteit van het licht met 10% toe of afnemen. De blacklight is toegevoegd aan het package. De installatie staat in bijlage L.

4.2.1.7 Camera-node

De camera bestaat uit twee onderdelen. Net zoals de lidar wordt de data van het camerabeeld vanaf de Raspberry Pi gepubliceerd op een rostopic /Image. Op de NUC draait een node die hierop subscribe en het live beeld weer geeft. De camera nodes op de Raspberry Pi en de NUC starten op bij het aanroepen van de algemene launch-file. De camera node zit in het package waarvan de installatie terug te vinden is in bijlage L.

4.2.1.8 Snapshot-node

De Snapshot-node zorgt ervoor dat foto's gemaakt worden van de camera beelden. Deze snapshots worden ook opgeslagen. Door op een knop te drukken wordt er een foto gemaakt en opgeslagen. De snapshot node start op met de camera node. De snapshot node zit in het package waarvan de installatie terug te vinden is in bijlage L.

4.2.1.9 Gyro-node

De gyro wordt uitgelezen door de Raspberry Pi pi. Bij het starten van de launchfile op de Raspberry Pi wordt deze node gestart. De data van de gyro wordt gepubliceerd op de /gyroAxes topic. Deze data bestaan uit een array van 3 floating point getallen. De gyro node zit in het package waarvan de installatie terug te vinden is in bijlage L.

4.2.2 Raspberry Pi shield

De nieuwe Raspberry Pi shield is ontworpen in een online pcb-tool EasyEDA. Hierin kan eenvoudig met bestaande componenten een PCB worden ontworpen en verstuurd naar JLCPCB voor productie. Voordat componenten op een PCB kunnen worden geplaatst dient eerst een schema te worden gemaakt waarop alle verbindingen tussen de componenten zijn aangegeven.

4.2.2.1 Spanningsvoorzieningen

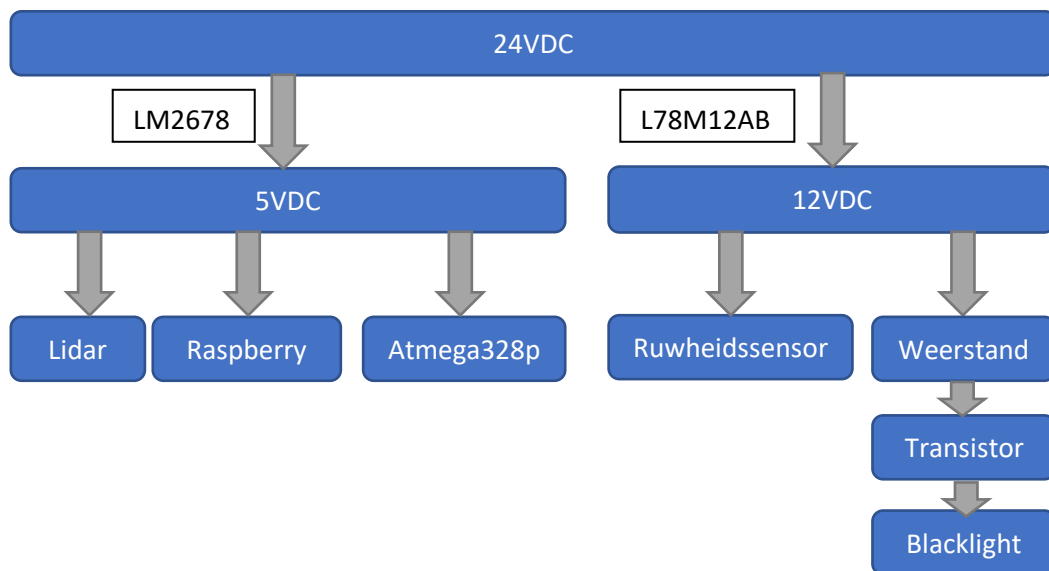
Om de componenten op en aan de PCB van stroom te voorzien is er een schema gerealiseerd van de voeding op de PCB. In het ontwerp is gekozen voor een LM2678 converter die de aanwezige 24 volt efficiënt verlaagd naar 5 volt gelijkspanning. Om de converter de gewenste 5 volt aan de uitgang te laten leveren dienen er een aantal componenten te worden toegevoegd. Daarvoor is in de datasheet van de LM2678 geraadpleegd. Hierin staat beschreven dat voor een 5 volt spanning er de volgende componenten moeten worden toegevoegd:

- Twee condensatoren van 1200 uF op de ingang
- Een spoel van 22uH
- Een diode
- Feedback condensator 0.01 uF
- Een condensator 22 uF op de uitgang

Deze componenten zijn vervolgens volgens de datasheet getekend in het schema. Om de bestaande 24 volt voeding op de pcb aan te kunnen sluiten zijn 2 klemmen toegevoegd. Daarnaast is op de uitgang van de converter een groene led aangesloten om een indicatie te geven dat de PCB aan staat. Vanaf de uitgang van de converter worden de Lidar, Raspberry Pi en de Atmega328p van voeding voorzien. Om de Lidar op de PCB aan te kunnen sluiten is in het schema een aansluitklem gezet op de uitgang van de converter.

De ruwheidssensor en de blacklight-leds werken op een lagere spanning dan 24 VDC. De ruwheidssensor werkt op 12 Volt en de blacklights werken op 7 Volt. Om de binnenkomende spanning van 24 volt te verlagen naar 12 volt is er een lineaire spanningsregelaar L78M12AB in het schema geplaatst. De uitgang van de spanningsregelaar is op twee klemmen aangesloten waarom de ruwheidssensor en de blacklight leds kunnen worden aangesloten. ou

De blacklights werken op een lagere spanning dan 12 volt. De spanning is daarom verlaagd met een voorschakelweerstand om de gewenste spanning van 7 volt te krijgen. Deze is berekend op 22 Ohm. De berekeningen zijn terug te vinden in bijlage F. Naast de weerstand is een 2N3904 transistor in het blacklightcircuit toegevoegd om de leds te dimmen en uit te schakelen. Afhankelijk van het camerabeeld kan de blacklight worden afgesteld zodat het optimale beeld kan worden verkregen. Hieronder in figuur 11 is een overzicht afgebeeld van de spanningsvoorziening op de PCB.



Figuur 11 spanningsboom

4.2.2.2 Sensoren

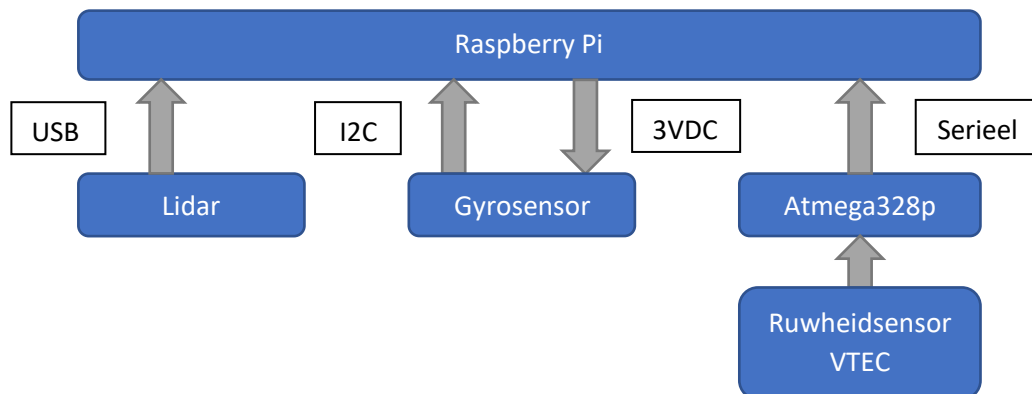
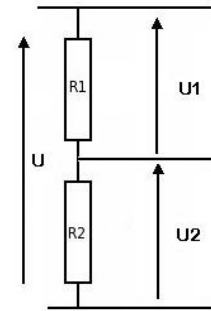
De inspectierobot is voorzien van verschillende sensoren. Deze dienen te worden verbonden op de Raspberry Pi via de PCB shield. Elke sensor heeft een eigen communicatieprotocol. In tabel 7 is een overzicht weergegeven van alle sensoren met communicatieprotocol.

Tabel 7 sensor protocollen

Sensor	Protocol
Lidar	USB
Gyro	I2C
Ruwheidssensor/Atmega328p	Serieel

Het nieuwe ontwerp bevat een lidar en gyrosensor. De lidar wordt via USB direct op de Raspberry Pi aangesloten. De gyrosensor maakt gebruik van het I2C protocol. Dit protocol is speciaal ontwikkeld voor communicatie tussen componenten op een PCB. Op de Raspberry Pi zijn aansluitingen aanwezig waar componenten met I2C kunnen worden aangesloten. Naast de I2C verbinding naar de Raspberry Pi, wordt de gyrosensor vanaf de Raspberry Pi van spanning voorzien. In onderstaande afbeelding is het overzicht weergegeven.

De Atmega328p beschikt over twee aansluitingen voor het ontvangen en verzenden van seriële data. Deze twee signalen worden door middel van een 5 volt spanning overgestuurd. De Raspberry Pi beschikt over twee seriële aansluitingen die werken op een lagere spanning van 3.3 volt. Om dit spanningsverschil tussen beide componenten op te lossen is er een spanningsdeler opgenomen in het PCB-schema. Omdat de Raspberry Pi een lagere spanning signaal naar de Atmega328p stuurt, is de spanningsdeler enkel geplaatst naar de binnenkomende data vanuit de Atmega328p naar de Raspberry Pi. In figuur 12 is een spanningsdeler weergegeven. De totale spanning over de twee weerstanden bedraagt 5 volt. De gewenste 3,3 volt dient over weerstand U2 te vallen. Als deze spanning van de totale spanning wordt afgehaald betekend dat over weerstand U1 1,7 volt komt te staan. De verhouding tussen deze twee spanningen is gelijk aan 0.5. De weerstanden op de PCB dienen dezelfde verhouding te hebben om de gewenste spanning te krijgen naar de Raspberry Pi. In het schema zijn een weerstand van 3.3k ohm en 1.6k ohm geplaatst. De schreven verbindingen zijn weergegeven in het blokschema hieronder. De verbindingen tussen de Raspberry Pi en de sensoren zijn verwerkt in het PCB-schema in bijlage G.



Figuur 12 sensor structuur

4.2.2.3 Motoraansturing

In het ontwerp van de robot zijn de motordrivers van de motoren verbonden aan de PCB shield van de Raspberry Pi. Uit de analyse is gebleken dat elke motordriver 3 verschillende pinnen heeft voor de draairichting, inschakelen en snelheid. In het ontwerp zijn deze pinnen aangesloten op de GPIO van de Raspberry Pi. In onderstaande overzicht is aangegeven waar welke motorpin is aangesloten op de GPIO van de Raspberry Pi.

Naast de aansturing van de motordrivers is de shield voorzien van een Analog Digital Converter (ADC). Deze chip maakt het mogelijk om analoge signalen van de motordrivers te meten. De drivers kunnen een analog signaal sturen van de actuele motorstroom en de motorsnelheid. Om de analoge waarden uit te lezen op de Raspberry Pi is de chip verbonden met de SPI-aansluitingen van de Raspberry Pi. Dit is een communicatieprotocol om componenten te laten communiceren op een PCB. De aansluitingen van de ADC-chip zijn in tabel 8 aangegeven.

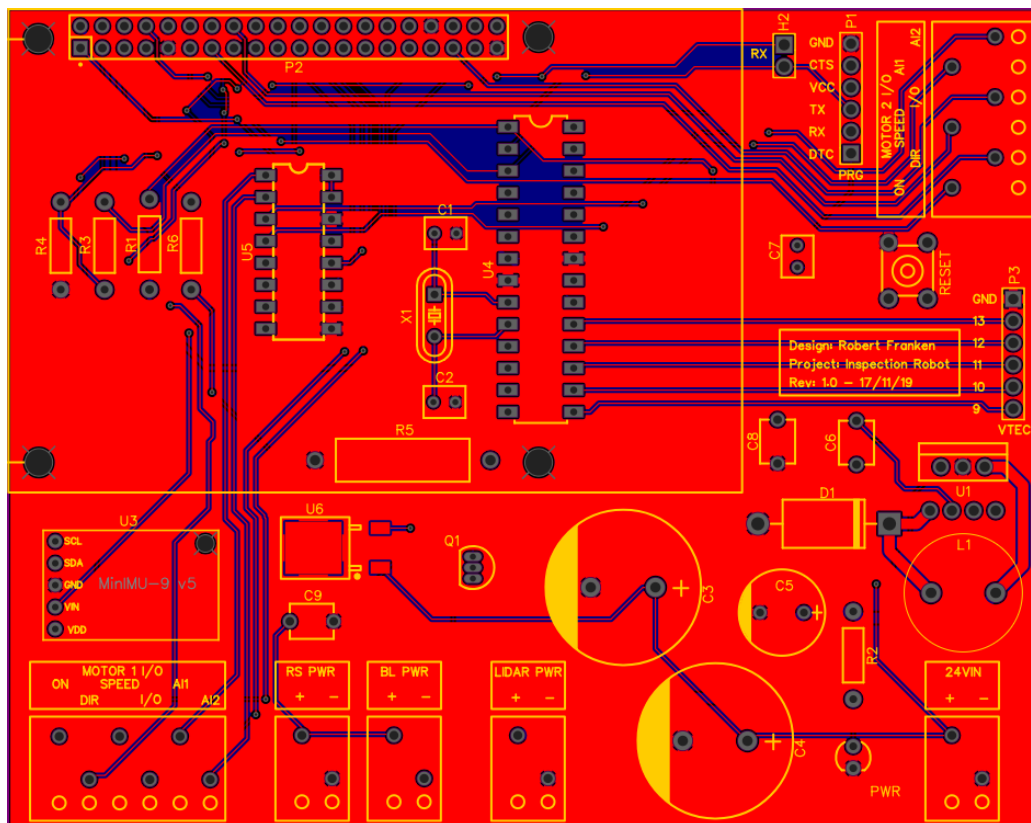
Tabel 8 Raspberry pin aansluitingen

Pin NR	Beschrijving	Aangesloten op
11	GPIO 17	Motor 2 Enable
13	GPIO 27	Motor 2 Direction

15	GPIO 22	Motor 1 Enable
16	GPIO 23	Motor 2 Speed PWM
18	GPIO 24	Motor 1 Direction
19	GPIO 10 SPI	ADC DIN
21	GPIO 9 SPI	ADC DOUT
23	GPIO 11 SPI	ADC CLK
24	GPIO 8 SPI	ADC CS
33	GPIO 13	GPIO Pin motor 1
35	GPIO 19	GPIO Pin motor 2
37	GPIO 26	Motor 1 Speed PWM
39	GND	ADC GND

4.2.2.4 Pcb-verbindingen

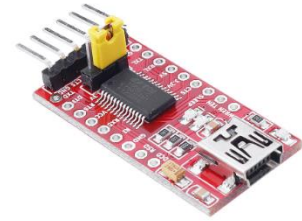
Nadat het schema van de PCB is gemaakt kunnen de componenten en verbindingen worden getekend op de PCB in EASY EDA. Hieronder in figuur 13 is de voltooide PCB afgebeeld.



Figuur 13 pcb-ontwerp

Met het plaatsen van de componenten is rekening gehouden met de hoogte van de componenten zodat deze de aansluitingen van de Raspberry Pi kunnen blokkeren. Daarnaast zijn voor de motordrivers, de ruwheidssensor, blacklight, lidar en de 24 volt voeding aansluitblokken geplaatst aan de rand van de PCB. Op deze blokken kunnen alle externe componenten worden aangesloten.

Om de Atmega328p chip te kunnen programmeren zonder dat deze losgehaald dient te worden is de pcb voorzien van een programmeerheader P3. Hierop kan een TTL naar USB-module worden geplaatst om de microcontroller te programmeren. In figuur 14 is een TTL naar USB module afgebeeld.



Figuur 14 usb programmeur

5. Beheren van de inspectierobot

In dit hoofdstuk worden de belangrijkste systemen van de robot toegelicht om de robot te kunnen laten functioneren. Daarnaast bevat het hoofdstuk de testresultaten van alle software en hardware. Tot slot wordt in stappen toegelicht hoe de robot kan worden opgestart en in gebruik kan worden genomen.

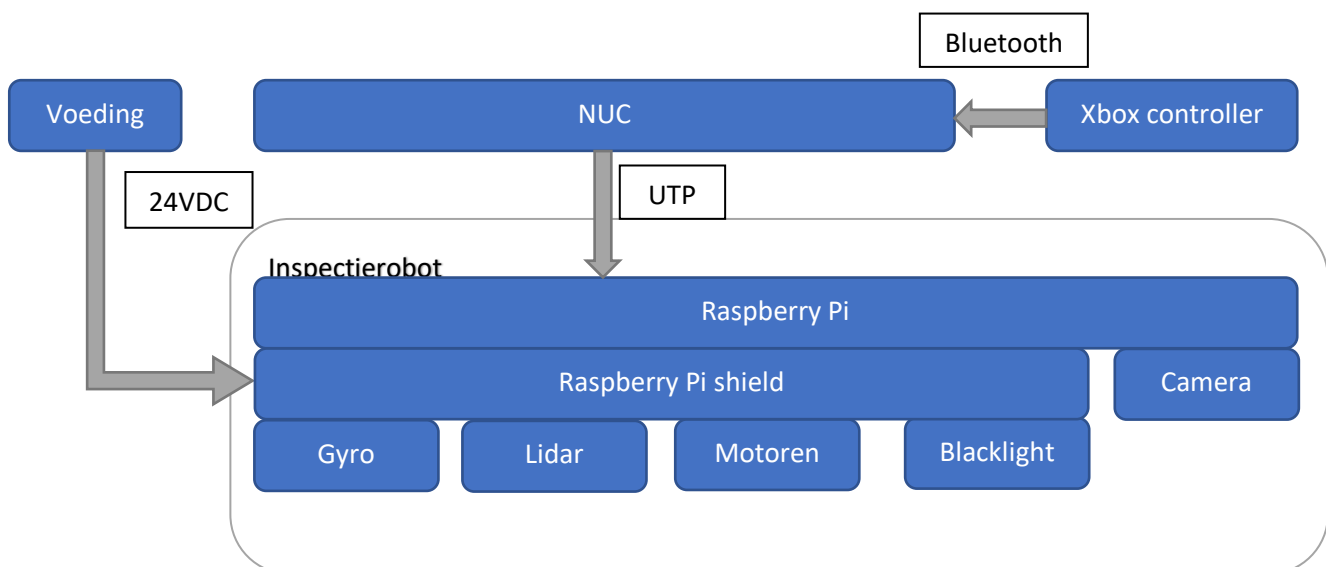
5.1 Robot systeemonderdelen

De inspectierobot bestaat uit drie verschillende onderdelen. Dit zijn de NUC, Raspberry Pi en de Xbox controller. Om de robot te starten zijn echter maar twee onderdelen nodig. Dit zijn de NUC en de Raspberry Pi. De Raspberry Pi wordt gebruikt om de robot aan te sturen. De NUC wordt gebruikt als interface van de robot. Door de ethernet kabel van de robot met de ethernet naar usb omvormer aan te sluiten op de NUC kan de robot gestart worden.

Hoe de robot gestart moet worden is terug te vinden in Bijlage M. Wanneer de software op de NUC wordt opgestart zullen er verschillende vensters geopend worden. Deze vensters weergeven de sensordata van de robot. Er worden in totaal twee vensters geopend. Deze zijn voor de Gyro en de Lidar. Het camera beeld wordt pas zichtbaar als er beeld ontvangen wordt van de Raspberry Pi.

Wanneer de software op de Raspberry Pi opgestart wordt zullen er scripts voor alle nodes opgestart worden. Als de software op de NUC al is opgestart, zal na het opstarten van de Raspberry Pi data in zichtbaar worden. Het script van de motoren en blacklights worden samen met de sensornodes opgestart.

Door de controller in te schakelen wordt deze automatisch verbonden met de NUC en kan de robot worden bestuurd. In figuur 15 zijn de deelsystemen te zien en hoe deze met elkaar in verbinding staan.



5.2 Hardware test

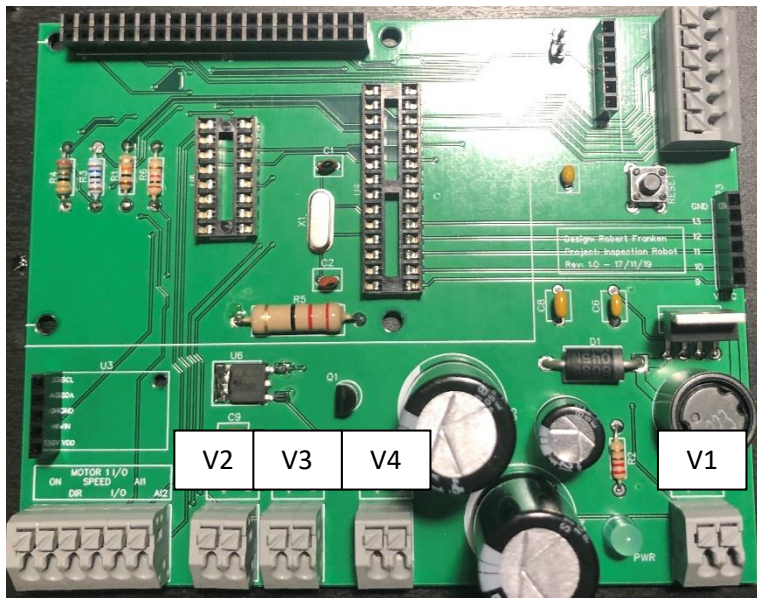
Figuur 15 deelsystemen

In deze paragraaf worden verschillende test scenario's beschreven en het gewenste resultaat.

5.2.1.1 Raspberry Pi Shield

Voor de nieuwe robot is een nieuwe PCB ontworpen en gerealiseerd. Om deze te controleren op juiste werking zijn er een aantal testen uitgevoerd. Voordat de shield aan kan worden gesloten zijn

een aantal metingen verricht. In figuur 16 is de PCB met aansluitingen weergegeven waarop wordt gemeten. Onder het figuur is in tabel 9 een overzicht gegeven van de metingen met resultaten.



Figuur 16 pcb aansluitingen

Voordat de PCB van spanning kan worden voorzien wordt de weerstand gemeten op V1, V2 V3 en V4. Als deze meting voldoet aan de gewenste waarde kan de PCB op V1 worden aangesloten op 24VDC. Vervolgens wordt op V2, V3 en V4 de spanning gemeten.

Tabel 9 Spanning test

Aansluiting	Type meting	Gewenste resultaat	Geslaagd
V1, V2, V3 en V4	Weerstand	> 1 mOhm	Ja
V2	Spanning	12 Volt +/- 5%	Ja
V3	Spanning	0 Volt	Ja
V4	Spanning	5 Volt +/- 2%	Ja

Nadat de testen zijn geslaagd zijn alle externe componenten aangesloten op de PCB shield volgens het overzicht in tabel 10

. De individuele sensoren en motoren worden in de volgende paragrafen getest met de ROS nodes.

Tabel 10 alle aansluitingen

Aansluitcode	Aansluiting	Naar component
M1	ON	Driver 1 – DI1
M1	DIR	Driver 1 – DI2
M1	SPEED	Driver 1 – AI1
M1	I/O	-
M1	AI1	Driver 1 – AO1
M1	AI2	Driver 1 – AO2

M2	ON	Driver 2 – DI1
M2	DIR	Driver 2 – DI2
M2	SPEED	Driver 2 – AI1
M2	I/O	-
M2	AI1	Driver 2 – AO1
M2	AI2	Driver 2 – AO2
V1	+	Voeding 24VDC +
V1	-	Voeding 24VDC -
V2	+	Ruwheidssensor +
V2	-	Ruwheidssensor -
V3	+	Blacklight led +
V3	-	Blacklight led -
V4	+	*Lidar +
V4	-	*Lidar -

*De Raspberry Pi is op de connector geplaatst en de shield is ingeschakeld. Na een aantal minuten is gebleken dat de 5 volt aansluiting op de GPIO van de Raspberry Pi niet geschikt is om de Raspberry te voeden vanaf de shield. Om dit op te lossen is de voeding van de Lidar op connector V4 losgehaald en aangesloten op de micro USB-aansluiting van de Raspberry. Om de Lidar van voeding te voorzien is ervoor gekozen om aparte voeding te plaatsen. Deze voeding zet de binnenkomende 24 volt om naar 5 volt.

5.3 Cleanrobot FAT

Tijdens de realisatie van de robot zijn alle individuele onderdelen van de robot getest. Aan het einde van het tijdsbestek is de robot in zijn geheel beproefd op de gewenste resultaten. Al deze gewenste resultaten samen met het antwoord of de test is geslaagd zijn te vinden in bijlage N.

Er is een onderdeel wat niet werkt. Dit is de ruwheidssensor. De ruwheidssensor lijkt softwarematig te werken, toch kan er niet bevestigd worden de data betrouwbaar is. De ruwheidsensor is verbonden met een Arduino, die weer verbonden is met de Raspberry. De link tussen de Arduino en de Raspberry is getest en ook door de test gekomen.

Uit de test blijkt de robot een status te hebben waarmee de hardware en software zijn voorbereid op autonomist software. De basis is er waarmee verder geborduurd kan worden.

5.4 Ingebruikname robot en aanbevelingen

De robot kan nog niet in gebruik genomen worden voor de taak waarvoor de robot wordt ontworpen. De status van de robot op moment van oplevering is een softwarematig geheel van de hardware. Dit wil zeggen dat de robot nog geen automatisch programma heeft, maar de basiselementen hiervoor nu wel aanwezig zijn. De stappen om de robot op te starten zijn beschreven in bijlage M.

6. Conclusie & aanbevelingen

Het doel van het verslag was om te onderzoeken welke aanpassingen en/of uitbreidingen noodzakelijk zijn om de huidige inspectierobot operationeel te maken. Daarvoor is de volgende hoofdvraag beantwoord: *Welke aanpassingen zijn benodigd om de huidige inspectierobot operationeel te krijgen?*

Om deze vraag te beantwoorden is onderzocht hoe de robot is opgeleverd door de voorgaande studenten. Daarvoor is de documentatie en de robot op software en hardware geïnspecteerd of alle beschreven onderdelen functioneren. Uit de inspectie is geconcludeerd dat de documentatie niet

overeenkomt met de actuele componenten in de robot. Daarnaast ontbreekt een duidelijke handleiding voor de software. Omdat verschillende componenten defect zijn en de software documentatie ontbreekt is ervoor gekozen om componenten te vervangen en de software herschrijven. Aan de hand van de aanpassingen is een nieuw ontwerp gerealiseerd om de robot operationeel te maken:

- Nieuwe PCB Raspberry Pi Shield
- Vervangen van de Raspberry Pi 3B
- Aanpassen van de motordriverinstellingen
- Vervangen van de ruwheidssensor door ander model/concept
- Nieuw robot besturingssysteem ROS

Naast de aanpassingen is de robot uitgebreid met de volgende sensoren en functionaliteiten:

- Besturing met Xbox controller
- Gyro sensor
- Lidar

De camera module samen met blacklights zijn nu ook bereikbaar en aanstuurbaar, hierbij zit een functie om een foto te kunnen nemen. De functie om de foto te nemen is daarentegen nog niet af. Door een druk op de knop wordt een foto genomen, bij de tweede keer een foto nemen wordt de oude foto overschreven met de nieuwe. Hiervoor kan een automatisch programma gemaakt worden die foto's classificeert en sorteert.

De controller is een robuuste toevoeging die uiteindelijk vervangen zal worden door een autonoom programma. Mocht er een fout optreden met het autonoom programma, kan de controller alsnog gebruikt worden om met het basissysteem de robot te besturen en of de robot handmatig verplaatst moet worden.

7. Bibliography

Batard, P. (2011 - 2020). *rufus*. Retrieved from <https://rufus.ie/>

Buzdar, K. (2020). Retrieved from vitux.com: <https://vitux.com/how-to-setup-dhcp-server-on-ubuntu/>

Hood, D. (2020). *installation ubuntu*. Retrieved from wiki.ros.org/: <https://wiki.ros.org/Installation/Ubuntu>

Ubuntu Team. (2020). *download*. Retrieved from <https://ubuntu.com/>: <https://ubuntu.com/download/>

Bronvermelding

(Ubuntu Team, 2020)

<https://ubuntu.com/download/>

(Batard, 2011 - 2020)

<https://rufus.ie/>

(Hood, 2020)

<https://wiki.ros.org/Installation/Ubuntu>

(Buzdar, 2020)

<https://vitux.com/how-to-setup-dhcp-server-on-ubuntu/>

Begrippenlijst

Woord	Betekenis
PWM	Pulse With modulation
API	Application programming interface
HTTP	Hyper tekst transfer protocol
SPA	Single page application
NUC	Next unit of computing
JSON	Javascript object notation

Symbolenlijst

Symbool	Eenheid	Beschrijving
Mechanisch		
m	[kg]	Massa
v	[m/s]	Snelheid
a	[m/s ²]	Versnelling
F	[N]	Kracht
g	[N/m]	Gravitatieversnelling
F_z	[N]	Zwaartekracht
F_{wr}	[N]	Wrijvingskracht
T	[Nm]	Torsie/Koppel
ω	[rad/s]	Rotatiesnelheid
N	[rpm]	Rotatiesnelheid
μ_s	[-]	Statische wrijvingscoëfficiënt
μ_d	[-]	Dynamische wrijvingscoëfficiënt
P_{mech}	[Watt]	Mechanische Vermogen
P_{eff}	[Watt]	Effectief vermogen
η	[-]	Rendement
i	[-]	Overbrengingsverhouding
i_{wiel}	[rad/m]	Overbrengingsverhouding wiel naar tandriem
Elektrisch		
I	[A]	Stroom
U	[V]	Spanning
A	[mm ²]	Aderdoorsnede
L	[m]	Lengte
U_{verl}	[V]	Acceptabel spanningsverlies
m_{km}	[kg]	Massa/km

Bijlage A – Plan van aanpak



2019

Plan van Aanpak

Het realiseren van een operationele inspectierobot

Auteurs:

Robert Franken (2109133)
Thom van Damme (2112137)
Thomas Brouwers (2118820)
Ewout de Gast (2123991)

Datum: 16-9-2019

Email:

rcm.franken@student.avans.nl
tjh.vandamme@student.avans.nl
tm.brouwers@student.avans.nl
epj.degast@student.avans.nl

Versie: 2.0

Document	Plan van Aanpak
Opleverdatum	-
Versie	2.0
Status	Ter beoordeling

Inhoudsopgave

1. Achtergronden	2
2. Projectopdracht	3
§2.1 Projectresultaten.....	3
§2.2 Pakket van Eisen.....	4
2.2.1. Functionele eisen.....	4
2.2.2. Niet functionele eisen.....	5
3. Aanpak	6
§3.1 Projectactiviteiten.....	6
§3.2 Mijlpalen.....	7
§3.3 Projectgrenzen.....	7
4. Beheersing	8
§4.1 Planning.....	8
§4.2 Projectorganisatie.....	8
§4.3 Faciliteiten.....	9
§4.4 Kwaliteit.....	9
§4.5 Kosten en Baten.....	9
5. Risico's	11
§5.1 Interne Risico's.....	11
§5.2 Externe Risico's.....	11
Bijlage A - Planning	12

Document	Plan van Aanpak
Opleverdatum	-
Versie	2.0
Status	Ter beoordeling

1. Achtergronden

In het Smart Tooling project worden bedrijven en serviceproviders als partner betrokken bij deelprojecten om technologie te ontwikkelen. Smart Tooling is een project binnen het Europese programma Interreg Vlaanderen-Nederland. Voor het project Robotica Smart Tooling ontwikkelt het KicMPi in samenwerking met partners uit Vlaanderen en Zuid-Nederland innovatie robotica toepassingen voor onderhoud in de procesindustrie. Gedurende de periode september 2018 tot en met januari 2019 heeft een voorgaande projectgroep meegewerkt aan het smart tooling project. Het deelaspect wat de studenten van Avans voor rekening namen is het bepalen van de oppervlakteruwheid van een chemische tank wand. Dit is noodzakelijk omdat het personeel dat de robot inzet niet met zekerheid kan stellen of de wand juist is schoongemaakt. Een nog belangrijker aspect is dat de regelgeving in de toekomst het niet meer toe om personeel te laten werken binnen een chemische tank. Daarom moet er een oplossing komen zodat de wand door het personeel geïnspecteerd kan worden zonder dat er een persoon daadwerkelijk de tank in moet. De vorige groep studenten hebben hiervoor een unit ontwikkeld die deze controle over kan nemen en kan automatiseren. Als de oppervlakteruwheid van de gehele wand is bepaald moet de robot deze informatie nog verwerken. Op basis van deze informatie moet de robot extra handelingen uitvoeren om alsnog de wand in zijn geheel te reinigen.

Document	Plan van Aanpak
Opleverdatum	-
Versie	2.0
Status	Ter beoordeling

2. Projectopdracht

De gebouwde inspectierobot is nog niet afgewerkt, software is vorig jaar geschreven om de robot wandruwheid te laten detecteren en ook met behulp van camera's olieresten te herkennen. De huidige robot is nog niet operationeel en dient verder afgewerkt te worden. Verder dient de robot zoveel mogelijk autonoom gaan functioneren. Het doel van het project is om de robot operationeel te maken zodat deze direct toegepast kan worden in een olietank.

§2.1 Projectresultaten

De opdrachtgever wenst aan het einde van de looptijd van het project een functionele robot als resultaat te zien. Omdat aan het eind van de eerste 10 weken het eerste resultaat dient te worden gepresenteerd, zijn de projectresultaten in 2 periodes opgedeeld:

Periode 1

1. Tussenverslag
2. Poster van het definitieve ontwerp
3. Materiaal en bestellijsten
4. Een inspectierobot die in staat is om:
 - Commando's voor bewegen te ontvangen
 - Sensordata te verzamelen en te versturen naar extern systeem
 - Een userinterface voor sensordata en commando's

Periode 2

1. Productverslag
2. PowerPointpresentatie
3. Video van het gerealiseerde product
5. Een inspectierobot die in staat is om:
 - Zich autonoom voort te bewegen
 - Verticaal omhoog te rijden
 - Zijn positie te bepalen

Document	Plan van Aanpak
Opleverdatum	-
Versie	2.0
Status	Ter beoordeling

§2.2 Pakket van Eisen

Voor het project zijn eisen opgesteld waar het eindresultaat aan dient te voldoen. Er is onderscheid gemaakt tussen functionele eisen en niet functionele eisen. Elke categorie bestaat uit een lijst met eisen en een prioriteit:

- **Must have** – Deze eis is noodzakelijk om het product te kunnen gebruiken.
- **Should have** – Deze eis is gewenst maar niet vereist om het product te laten werken.
- **Could have** – Aan deze eis wordt enkel voldaan mits er genoeg tijd is.
- **Would have** – Aan deze eis wordt waarschijnlijk niet voldaan en kan in de toekomst interessant zijn.

2.2.1. Functionele eisen

De functionele eisen zijn eisen die gesteld zijn aan de werking van het op te leveren systeem. In tabel 1 zijn deze eisen opgesteld.

Tabel 1 - Functionele eisen

Nr.	Eis	Prioriteit
1.	De oppervlakte ruwheid kan worden gemeten ISO 8501-1 norm.	Must have
2.	De kwaliteit van de oppervlakte kan worden beoordeeld d.m.v. camerabeeld en blacklight volgens de ISO 8501-1 norm.	Must have
3.	De data van de ruwheidssensor en de camerabeelden kunnen worden uitgelezen een extern systeem d.m.v. een UDP-verbinding	Must have
4.	De inspectierobot kan handmatig worden bestuurd d.m.v. een controller en/of UI van Serenity	Could have
5.	De inspectierobot kan autonoom door de tank rijden door gebruik te maken van sensoren	Could have
6.	De inspectierobot kan op eigen kracht verticaal omhoog rijden met een snelheid van 0.15 m/s	Must have
7.	De robot beschikt over een eddy-current sensor	Could have
8.	De robot beschikt over een LIDAR-sensor voor obstakeldetectie	Could have
9.	De robot beschikt over een camera voor het inspecteren van de oppervlakte	Must have
10.	De robot beschikt over een blacklight voor oppervlakte controle	Must have
11.	De inspectierobot moet voorzichtig omlaag rollen bij het uitvallen van de motoren.	Should have
12.	De inspectie data worden live gestreamd naar de interface van Serenity	Could have
13.	De robot beschikt over een ruwheidssensor	Must have
14.	De robot dient door een mangat te passen van 60 cm.	Must have
15.	De gegevens zullen worden opgeslagen worden in een database zodat er altijd een historie beschikbaar zal zijn	Could have
16.	Beeldcontrole vindt plaats op een server	Should have
17.	Dataverwerking van de sensoren vinden plaats op een externe server	Should have
18.	De server is extern bereikbaar om de data op te halen	Must have
19.	De gegevens zullen worden opgeslagen worden in een database zodat er altijd een historie beschikbaar zal zijn	Could have
20.	De rupsbanden zijn in staat om over gladde (olie) te rijden en grip te behouden	Could have

Document	Plan van Aanpak
Opleverdatum	-
Versie	2.0
Status	Ter beoordeling

2.2.2. Niet functionele eisen

Onder de niet functionele eisen vallen de overig geldende eisen. Deze eisen specificeren alle zaken die niet over de functionaliteit van het systeem gaan. In tabel 2 zijn deze eisen voor dit project opgesteld.

Tabel 2 - Niet functionele eisen

Nr.	Eis	Prioriteit
1.	De robot dient stofvrij en dompel vrij zijn (IP67)	Would have
2.	Alle componenten en draden moet beschermd zijn tegen aanraking (IP37)	Would have
3.	Het product moet voldoen aan de Machinerichtlijn 2006/42/EC	Would have
4.	Het product heeft een bevestigingspunt om zekeringslijnen aan te bevestigen	Would have
5.	Het systeem van de robot draait op Linux	Should have
6.	Het product is in staat om de locatie van vlekken op te slaan in een database via TCP/UDP.	Should have
7.	De documentatie bevat een solution architectuur	Must have
8.	De documentatie bevat een client architectuur	Must have
9.	De documentatie bevat een Raspberry PI-architectuur	Must have
10.	De documentatie bevat een NUC/server-architectuur	Must have
11.	De code bevat commentaar over de functie van de software	Must have
12.	Het systeem is plug en play. De robot is operationeel bij het aanzetten van de voeding.	Should have
13.	De onderdelen van de robot dienen eenvoudig te kunnen worden vervangen (< 10 handelingen)	Could have
14.	De robot heeft een massa van < 20 kg	Could have
15.	De robot dient in 10 minuten schoon te maken zijn na gebruik	Could have
16.	De robot dient voor 80% demontabel te zijn	Could have
17.	De onderdelen van de robot dienen voor 50% uit standaard onderdelen bestaan.	Could have
18.	De elektrische besturing binnen de robot bestaat uit 1 PCB.	Should have
19.	De robot is voorzien van externe werkverlichting/signalering	Could have

Document	Plan van Aanpak
Opleverdatum	-
Versie	2.0
Status	Ter beoordeling

3. Aanpak

§3.1 Projectactiviteiten

In dit hoofdstuk worden alle projectactiviteiten benoemd om de mijlpalen en eindresultaat te behalen. Deze worden vervolgens verwerkt in de planning.

Analyse

- Opstellen pakket van eisen
 - Vastleggen functionele en niet functionele eisen
 - Bespreken pakket van eisen met de opdrachtgever voor goedkeuring
- Opstellen plan van aanpak
- Onderzoek doen naar huidige status van de robot
 - Aansturing motoren (hardware/software)
 - Aansturing voor het uitlezen van de ruwheidssensor (hardware/software)
 - Aansturing van de camera, blacklight en ledverlichting (hardware/software)
 - De aansluitingen van de componenten in de robot (Hardware)
 - De opbouw van het besturingssysteem van de robot. (Software)
- Onderzoeken autonoom rijden
- Documenteren van de bevindingen in het productverslag

Ontwerp

- Opstellen ontwerpen van de deelsystemen
 - Software motor besturing
 - Software uitlezen ruwheidssensor
 - Software maken en verwerken van camera beelden, blacklight en ledverlichting
 - Software besturingssystemen inspectierobot/externe NUC
 - Ontwerp nieuwe componenten/aansluitingen
 - Berekening nieuwe type motoren
- Opstellen conceptontwerp autonoom rijden (hardware/software)
- Opstellen definitief ontwerp autonoom rijden (hardware/software)
- Opstellen van materiaallijsten
- Documenteren van de ontwerpen met een onderbouwing in het productverslag

Realisatie

- Realisatie van de (deel)systemen
 - Software schrijven motorbesturing
 - Software schrijven uitlezen ruwheidssensor
 - Software schrijven voor het maken en verwerken van camera beelden, blacklight en ledverlichting
 - Software schrijven voor de besturingssystemen inspectierobot/externe NUC
 - Alle deelsystemen samenvoegen aan het besturingssysteem van de inspectie robot
 - Software schrijven autonoom rijden
 - Installeren krachtigere motoren
- Documenteren van de gerealiseerde producten in het productverslag

Document	Plan van Aanpak
Opleverdatum	-
Versie	2.0
Status	Ter beoordeling

Beheer

- Oplevering van het eerste deel van het project
 - Testen van de motorbesturing
 - Testen van de ruwheidssensor
 - Testen camera, blacklight en ledverlichting
 - Testen besturingssysteem en alle deelsystemen
- Oplevering tweede deel van het project
 - Testen verticaal omhoog rijden
 - Testen positiebepaling/autonoom rijden
- Documenteren van de testresultaten in het productverslag

§3.2 Mijlpalen

Tijdens de looptijd van het project worden verschillende resultaten behaald. Door de verschillende mijlpalen voor het project te definiëren is de voortgang duidelijk en kan de planning indien nodig worden bijgestuurd. De mijlpalen van het project zijn te vinden in tabel 3.

Tabel 3 - Mijlpalen

Mijlpaal	Datum
Pakket van eisen	12-09-2019
Plan van aanpak	17-09-2019
Software ontwerp motorbesturing	12-10-2019
Software ontwerp oppervlakteruwheid sensor	12-10-2019
Software ontwerp camera beeldverwerking	12-10-2019
Het laten rijden van de inspectierobot	20-10-2019
Het uitlezen van de oppervlakteruwheid sensor	20-10-2019
Het maken van oppervlakte beelden	20-10-2019
Software ontwerp autonoom rijden	20-10-2019
Het vervangen van de motoren	18-11-2019

§3.3 Projectgrenzen

Onder projectactiviteiten zijn de werkzaamheden opgesomd die worden volbracht tijdens het project. In deze paragraaf worden de onderdelen beschreven die niet worden voldaan tijdens het project. Op deze manier heeft de opdrachtgever een duidelijk beeld wat er wel en niet wordt opgeleverd aan het einde van het project. De volgende onderdelen vallen buiten de grenzen van het project:

- Er wordt geen schoonmaakrobot opgeleverd voor het schoonmaken van de tank
- De constructie van de inspectierobot wordt niet aangepast/vernieuwd
- Voor de inspectierobot wordt geen onderhoud geleverd

Document	Plan van Aanpak
Opleverdatum	-
Versie	2.0
Status	Ter beoordeling

4. Beheersing

§4.1 Planning

Aan de hand van de projectactiviteiten is een planning opgesteld. Hierin zijn alle mijlpalen en deadlines in opgenomen. De planning is terug te vinden in bijlage A.

§4.2 Projectorganisatie

In deze paragraaf worden de verschillende rollen van iedere betrokkenen zijn van dit project. Voor elk persoon is hieronder een toelichting beschreven over de rol die hij of zij vervult binnen dit project.

Opdrachtgever/begeleider:

Jos van Kreij (am.vankreij@avans.nl)

Is het aanspreekpunt binnen het project als het gaat over de eisen die aan het eindresultaat worden gesteld. Daarnaast bewaakt de begeleider de voortgang van het project. Elke week wordt er vergaderd met het projectteam over de voortgang van het project zodat de opdrachtgever/begeleider op de hoogte is van de stand van zaken.

Teamleden

Naam	Rol
Robert Franken rcm.franken@student.avans.nl	Projectleider Contactpersoon VTEC
Thom van Damme tjh.vandamme@student.avans.nl	Planning Contactpersoon Nobleo
Thomas Brouwers tm.brouwers@student.avans.nl	Meedenkend lid
Ewout de Gast epj.degast@student.avans.nl	Scrum master

Document	Plan van Aanpak
Opleverdatum	-
Versie	2.0
Status	Ter beoordeling

§4.3 Faciliteiten

Om het project tot een succesvolle afronding te leiden zijn diverse faciliteiten nodig om de werkzaamheden uit te voeren. Deze faciliteiten zullen beschikbaar gesteld worden door Avans Hogeschool Breda. Het lokaal LD021 wordt als werkplek beschikbaar gesteld, op deze locatie worden werkzaamheden verricht aan de robot. Het lokaal LD023 inclusief de werkbanken worden beschikbaar gesteld voor gereedschap. Tevens is het mogelijk dat voor grote aanpassingen de faciliteiten in de werkplaats in gebouw LA beschikbaar worden gesteld.

§4.4 Kwaliteit

De kwaliteit van het project kan worden aangetoond door verschillende factoren. Er is een pakket van eisen opgesteld samen de opdrachtgever van Avans Hogeschool. Deze eisen zijn verdeeld in vier categorieën: Must-have, Should-have, Could-have en Wont-have. Op basis van de prioriteit van de eisen kunnen tussenresultaten wordt vastgelegd. Aan de mijlpalen worden deadlines ingepland na inschatting van benodigde tijd. Met deze combinatie heeft de opdrachtgever de mogelijkheid het project nauwkeuriger te volgen.

De eisen en tussenresultaten worden bijgehouden met de Scrum methode. Deze methode wordt toegepast om voornamelijk binnen het projectteam de taken en vooruitgang in beeld te brengen en te waarborgen. Het scrum bord is via de onlineapplicatie Trello met elk teamlid gedeeld. Buiten het project wordt er samengewerkt met een aantal bedrijven. De bedrijven en het projectteam komen maandelijks samen voor een vergadering. Hierin wordt de stand van zaken met elkaar gedeeld en op elkaar afgestemd. De bedrijven kunnen ook ingezet worden als hulp en informatiebron. Met de opdrachtgever wordt wekelijks vergaderd.

Het project wordt volgens het engineeringproces aangepakt. Dit proces wordt vastgelegd in documentatie. De documentatie wordt met de opdrachtgever besproken voor de deadlines. Samen met een goede communicatie tussen de opdrachtgever en tussen de teamleden zelf, wordt de kwaliteit hooggehouden.

§4.5 Kosten en Baten

Kosten

Het project is hardware matig grotendeels al gerealiseerd. De kosten van het project zitten in het verbeteren en uitbreiden van de inspectierobot. Voornamelijk wordt aan de software voor de besturing gewerkt van het project. Het ontwikkelen van de software voor de robot brengen geen extra kosten in het project. Bij het opstellen van dit document is nog niet duidelijk wat de exacte kosten zijn. De verwachte kosten van het project zijn nieuwe componenten voor het verbeteren en uitbreiden van de robot. In tabel 4 zijn de verwachte kosten van de componenten beschreven.

Tabel 4 - Kosten componenten

Component	Kosten
Tweetal stappen/servo motoren	€400,00
Een geschikte driver voor de motoren	€150,00
Sensoren	€500,00
Rupsbanden	€400,00
Totaal	€1450,00

Document	Plan van Aanpak
Opleverdatum	-
Versie	2.0
Status	Ter beoordeling

Baten

De baten staan niet in directe relatie met het project. Het project is ontstaan omdat de regelgeving in de toekomst geen personeel toelaat in een olietank voor onderhoud. Naast de regelgeving kan het een bedrijf zoals BASF personeelskosten besparen als de robots zelfstandig door de tank bewegen. Hiervoor is minder personeel nodig en kunnen kosten voor het schoonmaken van de tank worden gereduceerd.

Document	Plan van Aanpak
Opleverdatum	-
Versie	2.0
Status	Ter beoordeling

5. Risico's

Het project heeft diverse interne en externe risico's die van invloed kunnen zijn op het eindresultaat van het project. Om de risico's te beperken worden ze indien mogelijk vroegtijdig in kaart gebracht, in dit hoofdstuk worden deze risico's beschreven. De risico's worden in 2 categorieën opgedeeld, interne en externe risico's.

§5.1 Interne Risico's

- **Groepsleden die ziek worden**
Dit is vervelend indien een groepslid ziek is zal hij dit zo vroeg mogelijk moeten communiceren met de rest van de groep, zodat de groep indien nodig hierop kan inspelen.
- **Groepsleden die te laat komen**
Dit is vervelend en kan diverse externe oorzaken hebben, de groep kan de persoon hierop aanspreken als het meerdere malen voorkomt. Indien de persoon na meerdere waarschuwingen zich niet aan de afspraken houdt, wordt de tutor ingelicht over verdere stappen om de deadline niet in gevaar te laten komen.
- **Groepsleden die opdrachten te laat afmaken**
Dit is zeer vervelend en kan grote gevolgen hebben voor de oplevering van het project, door concrete gevolgen af te spreken zal dit risico ingeperkt moeten worden.
- **Uitloop van planning**
Door in de planning eventuele uitloop alvorens te plannen kan de impact van op de andere activiteiten worden beperkt.
- **Kwijtraken van bestanden**
Dit kan worden voorkomen door bestanden regelmatig een back-up te maken van de bestanden of door bestanden op te slaan op de cloud.

§5.2 Externe Risico's

- **Bestellingen die te laat geleverd worden**
Door onderdelen tijdig te bestellen kan dit risico ingeperkt worden, als een leverancier grote vertraging heeft zal het projectteam hier geen invloed op hebben. Het team kan als het mogelijk is kijken naar alternatieven
- **Defecte onderdelen**
Door voorzichtig met de onderdelen om te gaan en alles voor gebruik te testen kan dit risico beperkt.
- **Privéomstandigheden**
Het is niet mogelijk dit risico te voorkomen, door extra uitloop tijd in te plannen is het mogelijk om met de rest van de groep het werk over te nemen.
- **Faciliteiten die niet beschikbaar zijn**
Door op tijd aan te geven en de faciliteiten te reserveren kan dit risico ingeperkt worden, indien er een dubbele reservering is zal het projectteam hier niks aan kunnen doen.

Document Plan van Aanpak
Opleverdatum -
Versie 2.0
Status Ter beoordeling

Bijlage A - Planning

Schoonmaak	Periode 1										Periode 2																
	Wk. 1 28-aug	Wk. 2 4-sep	Wk. 3 11-sep	Wk. 4 18-sep	Wk. 5 25-sep	Wk. 6 2-sep	Wk. 7 9-sep	Wk. 8 16-sep	Wk. 9 23-sep	Wk. 10 30-sep	Wk. 11 7-okt	Wk. 12 14-okt	Wk. 13 21-okt	Wk. 14 28-okt	Wk. 15 4-nov	Wk. 16 11-nov	Wk. 17 18-nov	Wk. 18 25-nov	Wk. 19 2-dec	Wk. 20 9-dec	Wk. 21 16-dec	Wk. 22 23-dec	Wk. 23 30-dec	Wk. 24 6-jan	Wk. 25 13-jan	Wk. 26 20-jan	
Standdaatium																											
Opstellen pakket van eisen																											
Verifiëren functionele en niet-functionele eisen																											
Bereiden pakket van eisen met de opdrachtgever																											
Opstellen plan van aanpak																											
Onderzoek naar huidige status van de robot																											
Aanwijzing motoren (hardware/software)																											
Aanwijzing voor het uitlezen van de ruwheidsensor (hardware/software)																											
Aanwijzing van de camera, backlight en ledverlichting (hardware/software)																											
De aansluitingen van de componenten in de robot (hardware)																											
De opbouw van het besturingssysteem van de robot. (software)																											
Berekening nieuwe type motoren																											
Documenteren van de bevindingen in het productverlag																											
Opstellen ontwerp van de deelsystemen																											
Software motor besturing																											
Software uitlezen ruwheidsensor																											
Software maken en verwerken van camera beelden, backlight en ledverlichting																											
Software besturingssystemen inspectiebot/externe NUC																											
Ontwerp nieuwe componenten/aansluitingen																											
Berekening nieuwe type motoren																											
Berekening nieuwe type motoren																											
Opstellen conceptontwerp autonoom rijden (hardware/software)																											
Opstellen definitief ontwerp autonoom rijden (hardware/software)																											
Opstellen van materiaallijsten																											
Documenteren van de ontwerpen met een onderbouwing in het productverlag																											
Deelsystemen																											
Software schrijven motorbesturing																											
Software schrijven uitlezen ruwheidsensor																											
Software schrijven voor aansturing camera beelden, backlight en ledverlichting																											
Software schrijven voor de besturingssystemen inspectiebot/externe NUC																											
Alle deelsystemen samenvoegen aan het besturingssysteem van de inspectie robot																											
Verbetere aansluitingen/componenten robot																											
Installeren nieuwe motoren																											
Aanpassen elektrische componenten autonoom rijden																											
Software schrijven autonoom rijden																											
Documenteren van de gerealiseerde producten in het productverlag																											
Oplevering van het eerste deel van het project																											
Testen van de motorbesturing																											
Testen van de ruwheidsensor																											
Testen camera, backlight en ledverlichting																											
Testen besturingssysteem en alle deelsystemen																											
Oplevering tweede deel van het project																											
Testen verticaal omhoog rijden																											
Testen positieafwijking/autonoom rijden																											
Documenteren van de testresultaten in het productverlag																											

Looppijd activiteit
Eventuele uitloop
Kwaliteitsaanpak
Opgeleverde activiteiten

Bijlage B – Test python script motoraandrijving

```
import RPi.GPIO as GPIO
import time
import math
import xbox

GPIO_M1_ON = 22 # Gpio pin to turn motor 1 on
GPIO_M1_DIR = 24 # Gpio pin for motor 1 direction
GPIO_M1_SPEED = 13 # Gpio pin for motor 1 speed (PWM)

GPIO_M2_ON = 17 # Gpio pin to turn motor 2 on
GPIO_M2_DIR = 27 # Gpio pin for motor 2 direction
GPIO_M2_SPEED = 18 # Gpio pin for motor 2 speed (PWM)

# Use GPIO numbers not pin numbers
GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)

# Set pins as output
GPIO.setup(GPIO_M1_ON, GPIO.OUT)
GPIO.setup(GPIO_M1_DIR, GPIO.OUT)
GPIO.setup(GPIO_M2_ON, GPIO.OUT)
GPIO.setup(GPIO_M2_DIR, GPIO.OUT)
GPIO.setup(GPIO_M1_SPEED, GPIO.OUT)
GPIO.setup(GPIO_M2_SPEED, GPIO.OUT)
# 90
# 0-----180
# 270

if __name__ == '__main__':
    joy = xbox.Joystick()
    pwm1 = GPIO.PWM(GPIO_M1_SPEED, 100)
    pwm2 = GPIO.PWM(GPIO_M2_SPEED, 100)
    pwm1.start(0)
    pwm2.start(0)

    while not joy.Back():
        if joy.A():
            GPIO.output(GPIO_M1_DIR, GPIO.HIGH)
            GPIO.output(GPIO_M2_DIR, GPIO.HIGH)
            pwm1.ChangeDutyCycle(80)
            pwm2.ChangeDutyCycle(80)
            GPIO.output(GPIO_M1_ON, GPIO.HIGH)
            GPIO.output(GPIO_M2_ON, GPIO.HIGH)
        elif joy.B():
            GPIO.output(GPIO_M1_DIR, GPIO.LOW)
            GPIO.output(GPIO_M2_DIR, GPIO.LOW)
            pwm1.ChangeDutyCycle(80)
            pwm2.ChangeDutyCycle(80)
            GPIO.output(GPIO_M1_ON, GPIO.HIGH)
            GPIO.output(GPIO_M2_ON, GPIO.HIGH)
        if not (joy.A() or joy.B()):
            GPIO.output(GPIO_M1_ON, GPIO.LOW)
            GPIO.output(GPIO_M2_ON, GPIO.LOW)
            pwm1.ChangeDutyCycle(0)
            pwm2.ChangeDutyCycle(0)

    joy.close()
    pwm1.stop()
    pwm2.stop()
```

Bijlage C – Aandrijving testrapportage

Op basis van de code van de vorige groep is er een script geschreven in Python waarin met een Xbox controleren de motoren kunnen aansturen. Door dit script direct op de Pi uit te voeren zonder tussen onderdelen was er een minimale afhankelijkheid van de verschillende onderdelen.

Om te beginnen hebben zij de aansluiten van het PDB naar de Pi en de motor drivers gecontroleerd, tijdens het inspecteren en openen van de robot waren er een aantal grondkabels los deze zijn opnieuw aangesloten op de juiste plek. Vervolgens is de Pi opgestart en via SSH het script met bijbehorende bibliotheken op de Pi geïnstalleerd. Aan de hand van deze setup () werden de motoren testen.



Figuur 4- Setup analyse motoren

Tijdens het uitvoeren van het script ging alleen motor1 aan en ging deze roteren, motor2 bleef uit en ging niet roteren. Vervolgens zijn de aansluitingen opnieuw gecontroleerd en de driver van motor1 en motor2 omgewisseld, hierdoor werd geconcludeerd dat de driver van motor2 niet defect was. Bij het opnieuw uitvoeren van het script ging alleen motor1 aan en roteren, opnieuw bleef motor2 uit en ging deze niet roteren. Omdat motor2 weer niet aan ging is de stroomtoevoer van deze motor gecontroleerd en vergeleken met motor1. De waardes waren bij allebei de motoren gelijk, behalve het pwm signaal. Na het doormeten van de GPIO pinnen van de Pi werd de conclusie getrokken dat de Pi geen signaal stuurt naar het 2^{de} pwm kanaal. Om te controleren of motor2 functioneert is het pwm signaal van motor1 door gelust naar motor2. Hierdoor gingen allebei de motoren aan en roteren. Een observatie tijdens het testen van de motoren was dat de motoren op maximaal vermogen een minimale stuwkracht creëerde.

Na het aansluiten van een andere Pi waarbij het pwm kanaal wel werkt, was het mogelijk om beide motoren aan te sturen en de robot rond te laten rijden.

Bijlage D – Testrapportage camera en blacklight leds

Camera op de Pi aangesloten en in ROS een script uitgevoerd hiermee was het mogelijk om een video stream te starten naar een client. Tijdens de test is geconstateerd dat de verbinding tussen de camera en de Pi niet stabiel is.

De blacklight leds zijn bevestigd op een print waarvan de sporen te volgen zijn. Er zijn op deze print verschillende pinnen te zien. Door spanning op deze pinnen te zetten kunnen de leds worden aangestuurd. Op de print zijn geen weerstanden te zien. Om de leds toch te testen is de spanning vanaf 0 langzaam verhoogt tot een spanning van 8 volt en blijkt door onderzoek op het internet ook de werk spanning van deze leds te zijn. Er is verder geen aansturing voor de blacklights aanwezig in de inspectierobot.

Connectoren kabel camera module ver versleten en levert geen goeie connectie.

Bijlage E – Testrapportage ruwheidssensor

7.1.1 Uitgevoerde stappen

De sensor is geleverd door VTEC. De eerste stap was dus het lezen van documentatie die door VTEC gestuurd was. Er is een manual aangeleverd met installatie instructies en de output van de Arduino. De software heeft drie stappen.

1. Het algoritme scant het oppervlak en zoekt naar de maximale waarde. Het stuurt de hele scan van 4 kanalen en de som daarvan via serieel. Gebruikelijke uitgangswaarde - ongeveer 1,6 V. Zodra de lens op metaal gaat focussen, zou de output 1.8V of meer moeten bereiken. Als deze waarde niet is bereikt, betekent dit dat het metalen oppervlak te ver of te dicht bij de lens is. Probeer het apparaat te verplaatsen en begin opnieuw.

```

COM9
0.0000069560 -0.1165961742 -0.1165872097 -0.1165827512 -0.1165820837
-0.00000003304 -0.0647948074 -0.0647901010 -0.0647831535 -0.0647843599
-0.00000000124 -0.0306903147 -0.0306976397 -0.0306845784 -0.0306842446
0.00000000006 -0.0121759080 -0.0121740829 -0.0121730661 -0.0121728968
0.00000000000 -0.0036144301 -0.0036140799 -0.0036132102 -0.0036131258
-0.00000000000 -0.0003206169 -0.0003204804 -0.0003202499 -0.0003202761
-0.00000000000 0.0005937684 0.0005938210 0.0005938769 0.0005938988
0.00000000000 0.0006225720 0.0006225772 0.0006225719 0.0006225831
0.00000000000 0.0004237480 0.0004237171 0.0004237080 0.0004237145
-0.00000000000 0.0002349879 0.0002349823 0.0002349737 0.0002349708
-0.00000000000 0.0001110535 0.0001110458 0.0001110401 0.0001110455
0.00000000000 0.0000439666 0.0000439643 0.0000439593 0.0000439619
0.00000000000 0.0000129908 0.0000129896 0.0000129882 0.0000129883
-0.00000000000 0.0000010924 0.0000010919 0.0000010908 0.0000010907
-0.00000000000 -0.0000021936 -0.0000021938 -0.0000021940 -0.0000021941
0.00000000000 -0.0000022789 -0.0000022789 -0.0000022789 -0.0000022790
Autoscroll Show timestamp Newline 115200 baud Clear output
  
```

Figuur 9- Scannen naar maximale waarden over 4 kanalen

2. Het algoritme probeert dicht bij het doel te komen, dat is 95% van de maximale waarde die in stap één is gelezen. Geen seriële uitvoer tijdens deze fase.

3. Het algoritme probeert op de ingestelde waarde te blijven door de DAC-waarde aan te passen. Als het de focus verliest, wordt het oppervlak opnieuw gescand. Gedurende deze tijd geeft het de geschatte afstand van het hoofd tot het oppervlak in micrometer.

```

COM9
6832.6831054687
7036.5205078125
6951.6484375000
7011.4941406250
6970.5087890625
6986.8305664062
6966.1562500000
6967.6074218750
6955.2753906250
6950.9228515625
6941.8554687500
6935.6894531250
6927.7104492187
6921.1816406250
6913.5649414062
Autoscroll Show timestamp Newline 115200 baud Clear output
  
```

Figuur 10- Afstand tot het oppervlak

De data van de output is niet overeengekomen met de verwachte waardes. Er kwam ook geen logische verandering in de output wanneer de sensor verplaatst werd. Er moest dus uitgezocht worden waar de fout zat. Als eerste werd de code op de Arduino die meegeleverd was onderzocht.

Het was mogelijk dat de vorige projectgroep de code van VTEC aangepast had. Dit bleek niet zo te zijn en de code op de Arduino klopt ook.

Vervolgens is er naar de cd-lezer gekeken. Met behulp van een datasheet van de laser is de bedrading tussen de cd-lezer en de pcb onderzocht. De bedrading is in orde maar wanneer er stroom op het systeem gezet wordt brandt de laser niet. Om te kijken waar het probleem ligt is stroom geleverd aan de laser apart. Na deze test deed de laser het ook niet. De lens van de cd-lezer beweegt wel als er stroom op staat dus het probleem ligt bij de laser.

Bijlage F – Motor berekeningen

In de afbeelding x hiernaast is de huidige overbrenging situatie afgebeeld. De opstelling bestaat uit een motor, planetaire tandwiel overbrenging en een tandriem overbrenging. De motor levert een nominaal koppel van 0,128 N*m. De overbrengingsverhoudingen zijn in de afbeeldingen weergegeven. Met deze gegevens kan het nominale koppel worden berekend na de planetaire tandwieloverbrenging met onderstaande formule:

$$T_1 = T_m \times i$$

Waarin:

T_1 = Koppel na de eerste tandwiel overbrenging [N*m]

T_m = Geleverd koppel door de motor [N*m]

i = tandwielverhouding

Geeft de volgende berekening:

$$T_1 = 0,128 \times \frac{91}{1} = 11,65 \text{ N} * M$$

Vervolgens kan het berekende koppel worden gebruikt om de het koppel op de rupsband te berekenen met dezelfde formule en de overbrengingsverhouding van de tandriem:

$$T_2 = 11,65 \times \frac{2}{1} = 23,3 \text{ N} * M$$

De rupsbanden hebben een diameter van 0,125 meter. Met dit gegeven kan de kracht die de rupsband verticaal omhoog kan tillen worden bepaald met de formule:

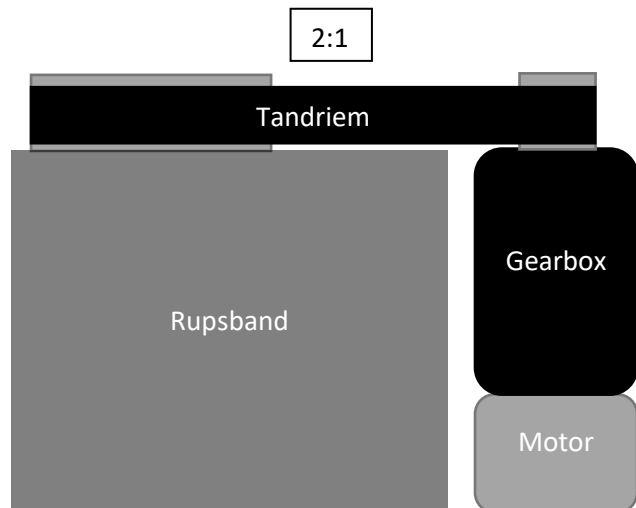
$$F_R = T_2 \times \frac{D}{2}$$

Waarin:

F_R = De kracht die de rups omhoog kan tillen [N]

D = De diameter van de rupsband [m]

$$F_R = \frac{23,3}{\frac{0,125}{2}} = 372,8 \text{ N}$$



Figuur 11: huidige overbrenging motor

Berekeningen vorige studentengroep

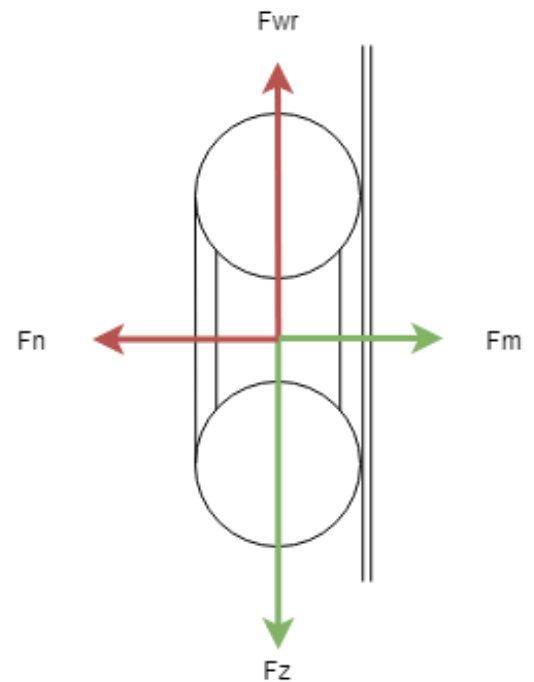
In deze bijlage zie je de situatieschets van de robot terug. Op basis van deze situatieschets zijn de motorberekeningen gemaakt en is uiteindelijk een motor gekozen.

Het gaat hierbij om een unit van 15 kg die met rupsbanden verticaal omhoog op een metalen wand rijdt.

De situatieschets voor het statische model is weergegeven in figuur 1 aan de rechterkant. Het betreft hierbij een magnetische crawler van 15 kg die verticaal omhoogklimt door gebruik te maken van rubberen rupsbanden op een metalen wand. Om te bepalen wat een geschikte motor is voor de unit worden een aantal parameters aangenomen op basis van de eisen van de klant.

$$m = 15 \text{ kg}$$

$$v = 0.15 \left[\frac{\text{m}}{\text{s}} \right]$$



Figuur 12: Statisch model

Statisch model

Om de crawler stil te laten staan moeten alle resulterende krachten gelijk zijn aan nul. Hierbij moet de zwaartekracht worden gecompenseerd en de normaalkracht die de wand uitoefent op de crawler. Dit is weergegeven door de volgende formules. Hierbij worden de krachten in de horizontale richting weergegeven door x en de krachten in de verticale richting door y.

$$\sum F_x \rightarrow F_{mag} - F_n = 0$$

$$F_{mag} = F_n$$

$$\sum F_y \rightarrow F_{wr} - F_z = 0$$

$$F_{wr} = F_z$$

Zoals weergegeven in de mathematische vergelijkingen moet de magneetkracht gelijk zijn aan de normaalkracht om de resultante in de x richting gelijk te stellen aan nul. De resultante in de y-coëfficiënt wordt gelijkgesteld aan nul door de zwaartekracht gelijk te stellen aan de wrijvingskracht.

Om te voorkomen dat de robot zal vallen moet er in de verticale richting moet de wrijvingskracht groter zijn dan de zwaartekracht. Hierdoor is gekozen om een bedrijfsfactor van anderhalf [s] toe te passen. Hierdoor zal de uiteindelijke wrijvingskracht altijd groter zijn dan de zwaartekracht waardoor de robot niet van de wand valt, dit is weergegeven door de volgende formule.

$$F_z = m * g = 15 * 9.81 = 147.15 \text{ [N]}$$

$$F_{wr} = F_z * s = 147.15 * 1.5 = 220.73 \text{ [N]}$$

$$F_z < F_{wr}$$

$$F_{Totaal} = 147,15 + 220,73 = 367,15 \text{ N}$$

Bijlage H – Installatie Ubuntu op NUC

Stap 1 - Download de ubuntu ISO vanaf internet of klik op de link:

<https://ubuntu.com/download/desktop/thankyou?country=NL&version=18.04.3&architecture=amd64>

Stap 2 - Download en installeer vervolgens het programma rufus:

<https://rufus.ie/>

Stap 3 - Gebruik een usb stick van minimaal 8gb en steek deze in uw pc en open vervolgens Rufus.

Stap 4 - Doorloop vervolgens de volgende stappen in Rufus:

1. Selecteer uw usb stick
2. Zoek en selecteer de ubuntu iso
3. Druk start en wacht tot het schrijven van de usb stick is voltooid

Stap 5 - na het schrijven kan de installatie usb in de NUC. Start nu de NUC op en druk op F2 om in het bootmenu te komen.

Stap 6 - Start de vanaf de installatie usb op

Stap 7 - Tijdens het opstarten volgt een keuze menu, selecteer install ubuntu en volg het installatie menu van ubuntu.

Ubuntu is nu op de NUC geïnstalleerd.

Bijlage I – Installatie ROS op Ubuntu

De installatie is te vinden op deze webpagina:

<https://wiki.ros.org/Installation/Ubuntu>

open de terminal en voer het volgende commando's stap voor stap in:

commando 1:

```
sudo sh -c 'echo "deb http://packages.ros.org/ros/ubuntu $(lsb_release -sc)
main" > /etc/apt/sources.list.d/ros-latest.list'
```

commando 2:

```
sudo apt-key adv --keyserver 'hkp://keyserver.ubuntu.com:80' --recv-key
C1CF6E31E6BADE8868B172B4F42ED6FBAB17C654
```

commando 3:

```
sudo apt update
```

commando 4:

```
sudo apt install ros-melodic-desktop-full
```

commando 5:

```
sudo rosdep init
```

commando 6:

```
rosdep update
```

commando 7:

```
echo "source /opt/ros/melodic/setup.bash" >> ~/.bashrc
```

commando 8:

```
sudo apt install python-rosinstall python-rosinstall-generator python-
wstool build-essential
```

de installatie is actief als bij het opnieuw opstarten van de terminal

Bijlage J – Verbinding controller met NUC

Open een terminal

Installeer sysutils met het commando:

```
sudo apt install sysfsutils
```

Open het bestand `/etc/sysfs.conf` en voeg onderaan het bestand deze regel toe:

```
/module/bluetooth/parameters/disable_ertm=1
```

Open het bluetooth menu met het commando:

```
sudo bluetoothctl
```

selecteer default agent:

```
default-agent
```

zet agent aan:

```
agent on
```

scan naar bluetooth apparaten:

```
scan on
```

zet de xbox controller aan door op de middelste Xboxknop te drukken zodat het lampje 1x per seconden knippert

druk nu de nieuwe verbinding knop in aan de voorzijde van de controller zodat het lampje 2x per seconden knippert

zoek in de terminal naar het macaderes van de xbox wireless controller en kopieer deze

```
9C:AA:1B:5C:36:2B
```

Zet de scan uit met het commando:

```
Scan off
```

Vertrouw de xbox controller met het commando:

```
trust 9C:AA:1B:5C:36:2B
```

controleer dat het indicatie lampje van de controller 2x per seconde knippert.

Paar de controller en de nuc met het commando:

```
pair 9C:AA:1B:5C:36:2B
```

Verbind met de xbox controller met het commando:

```
Connect 9C:AA:1B:5C:36:2B
```

Herstart nuc

Bijlage K – Installatie dhcp server

De stappen staan ook uitgelegd op de volgende website:

<https://vitux.com/how-to-setup-dhcp-server-on-ubuntu/>

Installeer de dhcp server met de volgende comando:

```
sudo apt-get install isc-dhcp-server -y
```

configureer de dhcp server met het commando:

```
sudo nano /etc/dhcp/dhcpd.conf
```

vul het gewenste netwerk in:

```
subnet 192.168.1.0 netmask 255.255.255.0 {  
    range 192.168.1.10 192.168.1.20;  
}
```

Controlleer in de configureer file de volgende instellingen:

```
default-lease-time 600;  
max-lease-time 7200;
```

Uncomment de volgende regel:

```
authoritative;
```

ga naar uw netwerkinstellingen en geef het interface voor de dhcp server een static ip adres:

```
192.168.1.10    255.255.255.0
```

sluit de config file en herstart de dhcp server met de volgende command:

```
sudo systemctl restart isc-dhcp-server.service
```

herstart het de netwerk service met het commando:

```
sudo systemctl restart NetworkManager.service
```

controleer de dhcp server met het commando:

```
sudo systemctl status NetworkManager.service
```

De dhcp server is nu active (running) zo niet ga bovenstaande stappen na.

Bekijk de dhcp leases met het commando:

```
dhcp-lease-list
```

Bijlage L – Installatie packages

Installeer pygame op de nuc met het commando:

```
sudo apt-get install python-pygame
```

Op de nuc installeer joy met het volgende commando:

```
sudo apt-get install ros-melodic-joy
```

Download de bijgeleverde packages op de nuc

```
scoutbot_ws.zip
```

Pak de zipfile uit en plaatst de map op gewenste locatie. Dit is het ros enviroment voor de robot

Als de map op gewenste locatie staat ga dan naar de home directory en open .bashrc :

```
sudo nano .bashrc
```

Zet onderaan .bashrc de volgende regel:

```
source YOUR_ENVIROMENT_PATH/scoutbot_ws/devel/setup.bash
```

Stel het ros ip van de nuc en de rosmaster in door onderaan .bashrc de volgende twee regels toe te voegen:

```
export ROS_IP="192.168.1.10"  
export ROS_MASTER_URI="http://192.168.1.10:11311"
```

Open een terminal of herstart de terminal

Bijlage M – Opstarten Robot

1 – Verbind de usb naar ethernet adapter met de nuc en sluit de ethernet van de robot hier aan.

2 – Steek de stekker van de robot in het stopcontact

3 – Druk op de aanknop van de NUC

4 - Log in op de NUC met het wachtwoord:

Wachtwoord: admin

5 - Open een terminal op de nuc (ctrl + alt + T) en start de roscore:

Commando: roscore

6 - Open een nieuw terminal en login op de raspberry via ssh:

Commando: sudo ssh [ubuntu@192.168.1.11](https://192.168.1.11)

Wachtwoord: ubuntu

6.1 - Wanneer er geen verbinding tot stand gebracht kan worden, herstart de dhcp server op de nuc:

Commando: sudo systemctl restart isc-dhcp-server.service

7 - Start de software op de raspberry:

Commando: roslaunch scoutbot startbot.launch

8 - open een nieuw terminal op de nuc en start de software:

Commando: roslaunch bot_master startmaster.launch

9 - verbind de controller door op de grote middelste Xbox knop te 2 seconden te drukken.

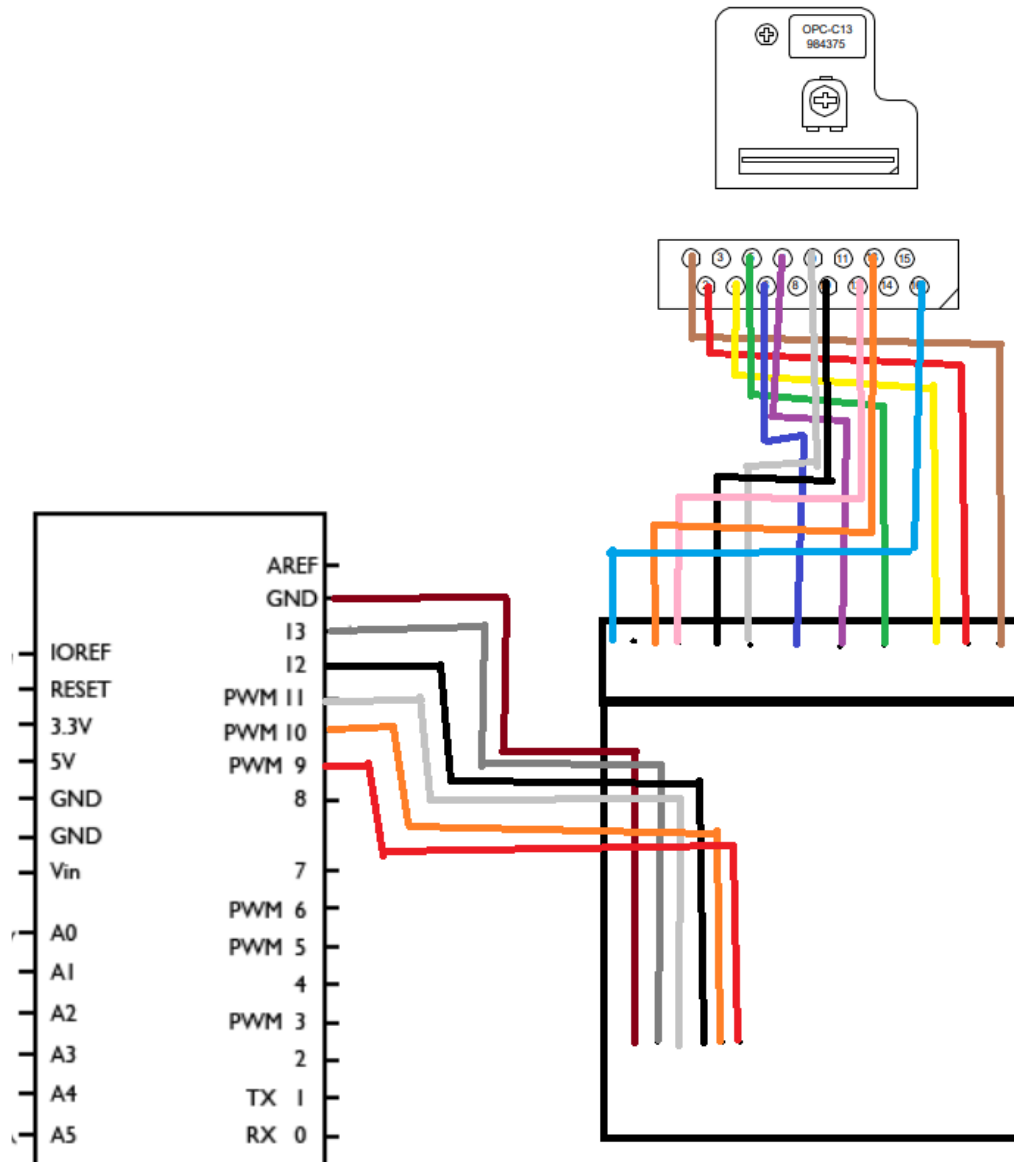
Have fun

Bijlage N – FAT		
Datum:	08-01-2020	
Tijdstip:	9:00	
Naam tester:	Robert Franken & Thom van Damme	
Studentnummer:	2109133 & 2112137	
Event/Response Nr:	1.1	
Test Run	Test input	Test geslaagd Ja/Nee met Opmerkingen
	De robot start op wanneer de stekker in het stopcontact wordt gestopt.	Ja
	De camera is verbonden met de raspberry en toegevoegd als node die gestart kan worden	ja
	Op de nuc kan een node gestart worden die het camerabeeld van weergeeft	ja
	De blacklight leds rond de cameras kunnen aan	ja
	De blacklight leds kunnen in felheid geregeld worden	ja
	Er kan een foto gemaakt worden van het camerabeeld en wordt opgeslagen wordt op de nuc	ja
	De gyro is verbonden met de raspberry en toegevoegd als node die gestart kan worden	ja
	De nuc kan de data van de gyro visueel weergeven	Ja – Het ziet er naar uit de gyro nog niet 100% goed werkt. De visualisatie lijkt een afwijking te hebben.
	De ruwheidsensor is als verbonden met de arduino	nee
	De arduino is verbonden met de raspberry	ja
	De arduino kan serieel data versturen naar de raspberry	ja
	De lidar is verbonden met de raspberry en toegevoegd als node die gestart kan worden	ja
	Op de nuc kan een node gestart worden die alle meetpunten van de lidar visueel weergeeft	ja
	De raspberry krijgt een ip adres van de nuc wanneer deze met de usb naar ethernet omvormer met elkaar verbonden zijn	ja
	Het is mogelijk om vanaf de nuc een ssh verbinding te maken met de raspberry	ja
	De roscore kan op de nuc gestart worden met het commando roscore	ja
	De software op de raspberry is te starten door een roslaunch commando aan te roepen	ja
	De software op de nuc kan gestart worden door 1 roslaunch commando aan te roepen	ja
	Bij het starten van alle software op de nuc komen alle visualisatie vensters naar boven	ja
	De xbox controller verbind na het indrukken van de Xboxknop automatisch met de nuc	ja

	Met de controller kan de robot kan vooruit rijden	ja
	Met de controller kan de robot kaachteruit rijden	ja
	Met de controller kan de robot kan sturen	ja
Conclusie Test succesvol doorstaan.		
Test geslaagd: Ja/Nee		

Bijlage O – Bedrading ruwheidsensor

In deze bijlage is de bedrading van de ruwheidsensor weergegeven. Hierbij is zowel de bedrading tussen de PCB en laser als de bedrading tussen de PCB en arduino te zien.



Figuur 12: Bedrading ruwheidsensor

Bijlage P – Code ruwheidsensor

```

#include <ros.h>

#include <ad7689.h>
#include <AD5752.h>

#include <Filters.h>
//create filters for the ADC
FilterTwoPole filterTwoLowpass_ch0;
FilterTwoPole filterTwoLowpass_ch1;
FilterTwoPole filterTwoLowpass_ch2;
FilterTwoPole filterTwoLowpass_ch3;
FilterTwoPole filterTwoLowpass_ch4;
FilterTwoPole filterTwoLowpass_dac;
uint32_t set_val = 0 ;

AD7689 *adc;
const uint8_t AD7689_SS_pin = 10;
uint8_t ch_cnt = 0; // channel counter

unsigned int min_val_dac = 25000; // -1.2V = 24903 -0.5V = 29491
//unsigned int max_val_dac = 31168; // 1.2V = 40632 0V=32768 -0.2V = 31457
unsigned int max_val_dac = 33168;
unsigned int val_dac = min_val_dac;

float max_ch0 = 0;
float max_ch1 = 0;
float max_ch2 = 0;
float max_ch3 = 0;
float max_ch4 = 0;

void setup() {
    // put your setup code here, to run once:
    Serial.begin(115200);
    while (!Serial);
    init_DAC(9, pn5V); //on this PCB there are some issues with initializing
the AD5752
    delay(500);
    init_DAC(9, pn5V);
    // AD7689 connected through SPI with SS specified in constructor
    // use default settings (8 channels, unipolar, referenced to 4.096V
internal bandga)
    adc = new AD7689(AD7689_SS_pin);
    delay(500);
    filterTwoLowpass_ch0.setAsFilter( LOWPASS_BESSEL, 4, 1.6);
    filterTwoLowpass_ch1.setAsFilter( LOWPASS_BESSEL, 1, 1.6);
    filterTwoLowpass_ch2.setAsFilter( LOWPASS_BESSEL, 1, 1.6);
    filterTwoLowpass_ch3.setAsFilter( LOWPASS_BESSEL, 1, 1.6);
    filterTwoLowpass_ch4.setAsFilter( LOWPASS_BESSEL, 1, 1.6);

    filterTwoLowpass_dac.setAsFilter( LOWPASS_BESSEL, 2, 0);
    while (val_dac < max_val_dac) {
        set_DAC_value(val_dac, DAC_AB);
        val_dac = val_dac + 5;
        Serial.println(val_dac, DEC);
        for (int i = 0; i < 20; i++) {
            filterTwoLowpass_ch0.input(adc->acquireChannel(p0, NULL));
            filterTwoLowpass_ch1.input(adc->acquireChannel(1, NULL));
            filterTwoLowpass_ch2.input(adc->acquireChannel(2, NULL));

```



```

    filterTwoLowpass_ch3.input(adc->acquireChannel(3, NULL));
    filterTwoLowpass_ch4.input(adc->acquireChannel(4, NULL));
}
//Debug output
Serial.print(filterTwoLowpass_ch0.output(), DEC);
Serial.print(" ");
Serial.print(filterTwoLowpass_ch1.output(), DEC);
Serial.print(" ");
Serial.print(filterTwoLowpass_ch2.output(), DEC);
Serial.print(" ");
Serial.print(filterTwoLowpass_ch3.output(), DEC);
Serial.print(" ");
Serial.println(filterTwoLowpass_ch4.output(), DEC);

//if measured val is bigger than previous max value
if (max_ch0 < filterTwoLowpass_ch0.output())
{
    max_ch0 = filterTwoLowpass_ch0.output();
}
if (max_ch1 < filterTwoLowpass_ch1.output())
{
    max_ch1 = filterTwoLowpass_ch1.output();
}
if (max_ch2 < filterTwoLowpass_ch2.output())
{
    max_ch2 = filterTwoLowpass_ch2.output();
}
if (max_ch3 < filterTwoLowpass_ch3.output())
{
    max_ch3 = filterTwoLowpass_ch3.output();
}
if (max_ch4 < filterTwoLowpass_ch4.output())
{
    max_ch4 = filterTwoLowpass_ch4.output();
}
}

}

void loop() {
    /*Serial.print(filterTwoLowpass_ch0.output(), DEC);
    Serial.print(" ");
    Serial.print(filterTwoLowpass_ch1.output(), DEC);
    Serial.print(" ");
    Serial.print(filterTwoLowpass_ch2.output(), DEC);
    Serial.print(" ");
    Serial.print(filterTwoLowpass_ch3.output(), DEC);
    Serial.print(" ");
    Serial.println(filterTwoLowpass_ch4.output(), DEC);
    // delay(10);*/
    val_dac = min_val_dac;
    float target = max_ch0 - (max_ch0 * 0.05) ;
    while (filterTwoLowpass_ch0.output() < target) //Second scan to find the
target value (first one, so left side of the parabola)
    {
        set_DAC_value(val_dac, DAC_AB);
        val_dac = val_dac + 10;
        if (val_dac > max_val_dac ) {
            val_dac = min_val_dac;
        } else if (val_dac < min_val_dac) {
            val_dac = max_val_dac;
        }
    }
}

```

```
    }

    for (int i = 0; i < 20; i++) {
        filterTwoLowpass_ch0.input(adc->acquireChannel(0, NULL));
    }
    delay(100);
    // Serial.println(filterTwoLowpass_ch0.output(), DEC);
}

while (1) {
    if (val_dac > max_val_dac ) {
        val_dac = min_val_dac;
    } else if (val_dac < min_val_dac) {
        val_dac = max_val_dac;
    }
    for (int i = 0; i < 50; i++) {
        filterTwoLowpass_ch0.input(adc->acquireChannel(0, NULL));
    }
    float filter_out = filterTwoLowpass_ch0.output();
    if ( filter_out < target)
    {
        val_dac = val_dac + 20;
    }
    if (filter_out > target)
    {
        val_dac = val_dac - 20;
    }
    set_DAC_value(val_dac, DAC_AB);
    filterTwoLowpass_dac.input(val_dac);
    //Serial.println(filterTwoLowpass_ch0.output(), DEC);
    uint32_t delta_dac = filterTwoLowpass_dac.output() - min_val_dac;
    float delta_dist = delta_dac * 0.3627 + 6614.7;
    Serial.println(delta_dist, DEC);
}
}
```