

Professionele bachelor Elektronica-ICT

Academiejaar: 2022 / 2023

Odisee
DE CO-HOGESCHOOL

Het maken en sturen van een robotarm

Lectoren: Sabine Martens
Mario Wyns

Verslaggevers: Lennert De Geeter
Roy De Volder
Jorik Van Dorpe
Lukas Verschraegen

Klas: 3ELO

Professionele bachelor Elektronica-ICT

Academiejaar: 2022 / 2023

Odisee
DE CO-HOGESCHOOL

Het maken en sturen van een robotarm

Lectoren:	Sabine Martens Mario Wyns
Verslaggevers:	Lennert De Geeter Roy De Volder Jorik Van Dorpe Lukas Verschraegen
Klas:	3ELO

Abstract

The creation and control of a robotic arm.

L. De Geeter, R. De Volder, J. Van Dorpe, L. Verschraegen

This work is an exhaustive exploration into the design, the creation and testing of a fully custom-built robotic arm with the objective of advancing our understanding and improving techniques in the field of robotics. The study commences with in-depth research into various types of motors, including stepper motors, servo motors, DC motors and brushless DC motors,

The research delves into the mechanical aspects of the project, examining the Cartesian robotic arm, the articulated robotic arm, and the SCARA robotic arm. The safety considerations are also analysed in detail. The design process is then illustrated, beginning with the creation of a block diagram, identification and detailed discussion of key components like the ESP32-S2, TMC2208, CP2102, ST7920, and rotary encoder. A comprehensive bill of materials (BOM) is provided, followed by a schematic representation and a pcb-design of the system that controls the robotic arm.

The realization section documents the assembly of the robotic arm, discussing the programming of the system and the integration of the motors. The report follows up with an extensive series of tests, including connectivity, safety measures, functionality, programming and mechanical parts.

The document concludes with an objective synthesis of the findings and their wider implications. Additional supporting materials, including copies of datasheets, the BOM, schematic diagrams and pcb-design, are included in the appendices. This investigation offers a detailed overview of the development and operation of a custom robotic arm system, providing a useful resource for future developments in the field of robotics.

Keywords: robotic arm system, pcb-design, 3D-design, schematic representation, assembly, functional testing, robotics, communication protocols

Inhoudsopgave

Figurenlijst	7
Tabellenlijst	9
Codefragmentenlijst	10
Afkortingenlijst	11
Begrippenlijst	13
Inleiding	14
1 Onderzoek	15
1.1 Motoren	15
1.1.1 Stappenmotor	16
1.1.2 Servomotor	16
1.1.3 DC-motoren	16
1.1.4 Brushless DC-motoren	16
1.2 Sensoren	19
1.2.1 Voeding	21
1.2.2 Communicatie	23
1.2.3 Programmatiecircuit	23
1.2.4 Sensor voor het aansturen van de stappenmotor	24
1.3 Mechanica	25
1.3.1 Cartesische robotarm	25
1.3.2 Articulated robotarm	26
1.3.3 SCARA-robotarm	27
1.4 Veiligheid	28
1.5 Opzetstuk	29
2 Ontwerp	31
2.1 Blokschema	31
2.2 Componenten	33
2.2.1 TMC2208	34
2.2.2 CP2102	35
2.2.3 ST7920	36
2.2.4 Rotary encoder	37
2.3 Mechanisch ontwerp	38
2.4 BOM (Bill of Materials)	40
2.5 Schema	41
2.6 Board design	42
3 Realisatie	43
3.1 Bestukking	43
3.2 Programmatie	45
3.3 Motoren	49
4 Testen	50
4.1 Connecties	50
4.2 Werking	51
4.3 Veiligheid	52

4.4	Functionaliteit	53
4.5	Programmatie	54
4.6	Mechanische onderdelen	55
	Conclusie	56
	Literatuurlijst	57
	Bijlagenoverzicht	60
	Bijlage 1: Kopieën datasheets	60
	Bijlage 2: BOM	63
	Bijlage 3: Schema	67
	Bijlage 4: Pcb	72
	Bijlage 5: Logboek rapporteren	73

Figurenlijst

Figuur 1: Arduino mkr wifi 1010 [6]	19
Figuur 2: ESP32 [7]	19
Figuur 3: ESP32-WROOM-32D chip zonder pcb [8]	21
Figuur 4: USB-C-pinout [9]	21
Figuur 5: Typische applicatie LDO [10]	22
Figuur 6: Full-step vs. micro-step [12]	24
Figuur 7: Verschillende types robotarmen [14]	25
Figuur 8: Cartesische robotarm en de aluminium profielen [16] [15]	25
Figuur 9: LumenPnP cartesische robotarm [17]	26
Figuur 10: Lasrobot [18] Figuur 11: Assemblagerobot [18]	26
Figuur 12: Scara-robotarm [15]	27
Figuur 13: Uitgewerkt versie PyBot [19] Figuur 14: 3D-Assenstelsel [19]	27
Figuur 15: Parallele grijper [36]	29
Figuur 16: Pneumatische hoek-grijper [37]	29
Figuur 17: Blokschema	31
Figuur 18: SparkFun Thing Plus - ESP32-S2 WROOM [21] Figuur 19: ESP32-S2 chip [22]	33
Figuur 20: Digikey TMC2208 board [23]	34
Figuur 21: CP2102 schema [24]	35
Figuur 22: ST7920 lcd [25]	36
Figuur 23: SPI pinnen [26]	36
Figuur 24: Rotary encoder schema [27]	37
Figuur 25: Rotary encoder op pcb [28]	37
Figuur 26: Uitwerking met geprinte vertanding	38
Figuur 27: Uitwerking met motoren	38
Figuur 28: Connectie tussen de assen met buizen	39
Figuur 29: Grijper	39
Figuur 30: Spoelen in een stappenmotor [29]	41
Figuur 31: Calculator voor baandikte [31]	42
Figuur 32: Stencil voor de pcb	43
Figuur 33: Pcb met soldeer pasta	44

Figuur 34: Werkend resultaat van de gesoldeerde pcb	44
Figuur 35: Volledige vierde as met servomontage en doorsnede met analyse	49
Figuur 36: TMC2208 risico	51
Figuur 37: Back-upplan	51
Figuur 38: Veiligheidsicoon hitte	52
Figuur 39: Eindresultaat robotarm	55
Figuur 40: Opsplitsen van een ontwerp	55
Figuur 41: Schema deel1	67
Figuur 42: Schema deel2 Bijlage 5	68
Figuur 43: Schema deel3 Bijlage 5	69
Figuur 44: Schema deel4 Bijlage 5	70
Figuur 45: Schema deel5 Bijlage 5	71
Figuur 46: Pcb	72

Tabellenlijst

Tabel 1: Afkortingenlijst	11
Tabel 2: Begrippenlijst	13
Tabel 3: Vergelijking van motoren [2] [3] [4] [5]	17
Tabel 4: Microcontrollers vergelijken [6] [7]	20
Tabel 5: BOM	63

Codefragmentenlijst

Codefragment 1: Toevoegen bibliotheken [32] [33]	45
Codefragment 2: Initialisatie stappenmotoren en servomotoren [32] [33]	45
Codefragment 3: Definiëren knoppen en variabelen aanmaken	45
Codefragment 4: Definiëren led en potentiometers en variabelen voor de potentiometers	46
Codefragment 5: Definiëren variabelen voor posities	46
Codefragment 6: Definiëren variabelen voor posities [32] [33]	47
Codefragment 7: Definiëren variabelen voor posities [32] [33]	47
Codefragment 8: Indrukken van de save knop	48
Codefragment 9: Indrukken van de save knop	48
Codefragment 10: Indrukken van de save knop [32] [33]	48
Codefragment 11: Indrukken van de stop knop	49
Codefragment 12: Indrukken van de save knop	54

Afkortingenlijst

Tabel 1: Afkortingenlijst

Afkorting	Betekenis
BOM	Bill of Materials
IC	Integrated circuit
DC	Direct current
PWM	Pulse-width modulation
MCU	Microcontroller unit
ESC	Electronic speed control
NEMA	National Electric Manufacturers' Association
pcb	Printed circuit board
IoT	Internet of Things
SoC	System-on-chip
SPI	Serial Peripheral Interface
I2C	Inter-Integrated Circuit
UART	Universal asynchronous receiver-transmitter
BLE	Bluetooth Low Energy
USB	Universal Serial Bus
LDO	Low-drop-out
TTL	Transistor-Transistor Logica
RMS	Root Mean Square

DIR	Direction
OTP	One-time programmable
SRAM	Static random-access memory
lcd	liquid crystal display
TMC	Trinamic
GPIO	General Purpose Input/ Output
elco	elektrolytische condensator

Begrippenlijst

Tabel 2: Begrippenlijst

Begrip	Verklaring
Brushless	Geen gebruik maken van fysieke contacten naar de commutator van de motor.
Flash	Flash is de verzamelnaam voor opslagsystemen waar data wordt opgeslagen op chips die hun informatie behouden nadat de voedingsspanning is weggehaald. Het is een vorm van niet-vluchtig geheugen dat in 1984 voor het eerst op de markt kwam.
SRAM	<p>Static random-access memory is een geheugen waarvan de inhoud niet periodiek hoeft geüpdatet worden.</p> <p>De opgeslagen data in een SRAM kunnen te allen tijde en in elke volgorde worden uitgelezen. Ze wijzigen enkel bij overschrijven of als de IC spanningsloos wordt. Het is dus een vluchtig geheugen.</p>
Pinout	Pinout is een verwijzing naar de pinnen of contacten die een elektrisch apparaat of connector verbinden. Het beschrijft de functies van uitgezonden signalen en de circuit input/output (I/O) vereisten.
Royaltyvrij	Royaltyvrij is een term die wordt gebruikt om bepaalde soorten intellectuele eigendom te beschrijven zonder royalty's te betalen. Eigenaars van intellectuele eigendom moeten dit label specifiek op hun inhoud toepassen, zodat iedereen het op dezelfde manier kan gebruiken.
Beheersmaatregelen	Maatregelen die worden genomen om de risico's waaraan ondernemingen blootstaan te beheersen.
Consciëntie	Nauwkeurigheid.
SCARA	Selective Compliance Articulated Robot Arm
Interferentie	Interferentie is de samen- of tegenwerking van verscheidene golven op dezelfde tijd en plaats.

Inleiding

In dit project wordt een prototype van een robotarm ontworpen en ontwikkeld. De arm moet met toegankelijke hardware en machines kunnen worden vervaardigd kunnen. De robotarm wordt ontworpen binnen een industriële omgeving waar hij repetitieve taken precies zal moeten uitvoeren.

In een eerste hoofdstuk wordt onderzoek gedaan naar de verschillende mogelijkheden van de arm. Die kan namelijk mechanisch uit heel wat verschillende ontwerpstrategieën bestaan. Ook kan afhankelijk van de toepassing en de grijpfunctionaliteit andere tooling nodig zijn. De robotarm wordt logischerwijze gestuurd door motoren, ook hier zijn weer een aantal mogelijkheden met voor- en nadelen. De robotarm is een prototype voor in een industriële omgeving, ook veiligheid is dus een aandachtspunt.

Eens de basiseigenschappen in het onderzoek zijn uitgeklaard, kan al deze kennis in het ontwerp worden overgenomen. Dat is dan ook het volgende hoofdstuk van dit onderzoek en omvat zowel een mechanisch als elektronisch ontwerp. Delen zoals een blokschema, componenten, Bill of Materials en elektronische schema's, alsook een finaal mechanisch ontwerp worden in dit tweede hoofdstuk besproken.

Eens alle ontwerpen en onderzoeken zijn uitgevoerd, kan er worden overgegaan tot de realisatie van de arm. In het derde hoofdstuk wordt de bestukking van de pcb besproken en de programmatie ervan. Ook het 3D-printen van de arm en zijn assemblageproces komen hier aan bod.

Wanneer de robotarm tot slot dan volledig geassembleerd is, kan de testfase worden ingezet. In dit vierde hoofdstuk worden functionaliteit en veiligheid extra onder de loep genomen en eveneens het oplossen van problemen met mechanische onderdelen en programmatie.

Via een experimentele onderzoeksmethode wordt dit project en dossier opgebouwd. De belangrijkste bronnen zijn datasheets van de componenten en diverse online bronnen en ontwerpen.

1 Onderzoek

Om vlot van start te gaan met het project is het belangrijk om voorkennis te hebben van de onderdelen van een robotarm.

In dit hoofdstuk worden verschillende onderdelen vergeleken die noodzakelijk zijn om een robotarm te laten functioneren. De focus ligt vooral op de realiseerbaarheid en het gebruiksgemak van verschillende producten. Vooraleer een ontwerp te realiseren, is het vergelijken van producten noodzakelijk om de haalbaarheid te controleren. Kosten en gebruiksgemak zijn hierbij uiteraard belangrijk.

De motoren, sensoren, het mechanische deel van de robotarm, de veiligheid en tot slot het opzetstuk van de robotarm zijn de belangrijkste onderdelen van dit project. Dit is de reden waarom er naar elk onderdeel onderzoek zal gebeuren. Om verschillende materialen te verplaatsen zijn immers andere en eventueel nauwkeurigere opzetstukken nodig. Hierdoor wordt het belang van de vervangbaarheid van een opzetstuk benadrukt.

1.1 Motoren

De hoofdzaak, en tevens de belangrijkste onderdelen van een robotarm zijn de motoren. De kern van een motor bestaat uit een koperen spoel met een massadichtheid van $8,96 \text{ kg/m}^3$. Koper in de motoren is erg zwaar, de keuze voor de juiste motor is dus belangrijk. [1]

Eerst wordt aan de basis geopteerd om een motor van 300 à 400 gram te gebruiken. Vervolgens gaat op het uiteinde van de robotarm de keuze naar een motor van 50 à 60 gram. Ten slotte is het doel dat het geheel van de motoren nog steeds krachtig genoeg is om 250 tot 500 gram op te tillen.

Bovendien is de vloeiende beweging van een motor belangrijk. Uiteraard is dit afhankelijk van de sensoren waarmee de motor wordt aangestuurd. Vandaar het gebruik van motoren die compatibel zijn met een IC. Om de kracht in de motor te behouden wordt onderzocht of er interne tandwielen zijn geplaatst.

Naast de vooraf besproken aspecten speelt ook de kostprijs, die varieert naargelang de grootte van de motor, een grote rol. Hoe groter de motor, hoe hoger de prijs is. Want hoe groter de motor, hoe meer koper er wordt gebruikt en dus ook de kostprijs die stijgt. Efficiëntie is erg belangrijk, maar minder van toepassing bij een motor door zijn mindere waarden op dit vlak.

Er bestaan verschillende motoren zoals: stappenmotoren, servomotoren, DC-motoren en brushless DC-motoren. De vergelijking en de uitleg van de meest voorkomende motoren in de elektronica staan in Tabel 3. Tabel 3: Vergelijking van motoren. De vergeleken motoren zijn geselecteerd op basis van populariteit en kracht.

1.1.1 Stappenmotor

Als er gekeken wordt naar kracht en populariteit, dan wordt er vaak gebruik gemaakt van een NEMA 17-motor. De motor is vooral populair in toepassingen met een 3D-printer [2].

1.1.2 Servomotor

Bij een servomotor is kracht een grote factor, een motor die 25 kg kan heffen is dus de beste keuze. De td-8125mg is een veel gebruikte optie en is hier de beste mogelijkheid [3].

1.1.3 DC-motoren

Voor DC-motoren wordt de term “transmissiemotor” gebruikt. Bij dit type motor wordt gebruik gemaakt van een tandwieloverbrenging. [4]

1.1.4 Brushless DC-motoren

Aangezien de brushless DC-motoren van oorsprong weinig gewicht kunnen dragen, is de populariteit van belang. Dronemotoren, zoals de Pichlermotor, zijn de meest gebruikte in hun soort [5].

Uit Tabel 3 kan geconcludeerd worden dat er in dit project een combinatie van stappenmotoren en servomotoren wordt gebruikt. De brushless DC-motor en de DC-motor hebben allebei een continue beweging en zijn dus minder geschikt voor deze toepassing

Tabel 3: Vergelijking van motoren [2] [3] [2] [3]

	Stappenmotoren [2]	Servomotoren [3]	DC-motoren [4]	Brushless DC-motoren [5]
Beweging	Een stappenmotor kan verschillende posities aannemen. Er is een vloeiende beweging om van de ene positie naar de andere te gaan. De verandering gebeurt heel snel en precies.	Een servomotor heeft een vloeiende beweging bij het verdraaien, maar is niet altijd even precies doordat sommige motoren plastic tandwielen hebben. Het voorbeeld is daarom met metalen tandwielen.	De vloeiende beweging van een DC-motor draait constant in eenzelfde richting. Voor de robotarm is de DC-motor niet geschikt aangezien er een positie moet worden bepaald.	Een brushless DC-motor is een motor die heel vloeiend draait, maar hij heeft ook een continue beweging zoals de normale DC-motor. Er is ook een speciale sturing nodig om controle te krijgen over de snelheid.
Gewicht motor	280 g	62 g	200 g	29 g
Hefkracht	4,0 kg/cm	25 kg/cm	75 kg/cm	730 g/cm
Kostprijs	€ 11	€ 11,5	€ 12	€ 27,99
Tandwielen	Er zijn geen ingebouwde tandwielen aanwezig.	De motor in het voorbeeld heeft ingebouwde, metalen en robuuste tandwielen om aan een groot hefvermogen te komen.	De motor in het voorbeeld heeft ingebouwde, metalen en robuuste tandwielen om aan een groot hefvermogen te komen.	Er zijn geen ingebouwde tandwielen aanwezig.

Draaihoek	Er is een staphoek van 1,8° per stap.	0-180° aangestuurd met een PWM-sigitaal.	Een DC-motor heeft altijd een continue draaihoek, en is dus niet instelbaar.	Een brushless DC-motor heeft altijd een continue draaihoek, en is dus niet instelbaar.
Stroom	1,7 A	3,4 A	200 mA	23 A
Aanstuurbaarheid	Aangezien er meerdere spoelen worden gebruikt in de motor is er een bepaalde IC nodig om ze aan te sturen.	De servomotor heeft de mogelijkheid om via PWM te worden aangestuurd. Dit is gemakkelijk omdat bijna elke MCU dit signaal kan genereren.	Deze motor kan worden aangestuurd door voeding op de motor te zetten, dit werkt zonder enig stuursigitaal.	Om aan te sturen heeft de brushless DC-motor een ESC nodig.
Aansluitbaarheid	Deze motor heeft een dikke en stevige as van 5 mm diameter om iets op aan te sluiten. Een veel gebruikte motor is een NEMA 17-motor met standaard montagepunten.	De motor heeft een metalen as met ribbels om betere grip te krijgen en een hol center waar een schroef kan worden ingedraaid. De aansluiting is dus heel sterk en stevig. Een nadeel is dat de diameter van de as redelijk klein is. Er zijn twee montagepunten aanwezig.	De aansluitbaarheid bij deze DC-motor is super, hij heeft een D-vormige as voor ultieme krachtoverbrenging. De asdiameter is 6 mm en de motor zelf heeft geen montagepunten.	De motor kan goed aan onderdelen worden gekoppeld, doordat hij een stevige metalen as heeft met een diameter van 15 mm. Hierop zit nog een bevestigingsstuk dat kan worden vastgeschroefd. De motor heeft vier montagepunten.

1.2 Sensoren

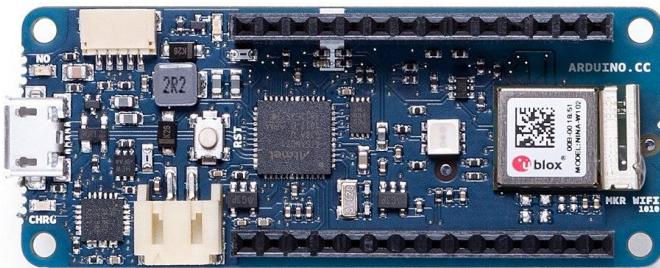
Sensoren zijn van groot belang bij een robotarm. Vooraleer de sensoren iets kunnen lezen of sturen, moet er een microcontroller worden gezocht. De MCU stelt het brein van de pcb voor en is extreem belangrijk.

Er bestaan verschillende modellen microcontrollers. Aangezien een wificonnctie belangrijk is om bijvoorbeeld data te sturen naar een website, zijn er enkele microcontrollers die gebruik maken van ingebouwde antennes. Ze worden gebruikt bij IoT-toepassingen (Internet of Things).

Er wordt een vergelijking gemaakt tussen de twee populairste MCU's van de grootspelers op de markt, namelijk Arduino met "Arduino mkr wifi 1010" Figuur 1 [6] en Espressif met "ESP32" Figuur 2 [7]. Tabel 4

Aangezien de ESP-boards open source zijn, zijn er ook andere fabrikanten die gelijkaardige system on a chip (SoC) microcontrollers verkopen.

De ESP-boards kunnen zowel voor professionele als amateurtoepassingen worden gebruikt. De boards zijn open source en op die manier gemakkelijk te achterhalen hoe ze kunnen worden geprogrammeerd.



Figuur 1: Arduino mkr wifi 1010 [6]



Figuur 2: ESP32 [7]

Tabel 4: Microcontrollers vergelijken [6] [7]

Fabrikant	Arduino [6]		Espressif [7]	
Product	Mkr wifi 1010		ESP32	
Chip	ATSAMD21		ESP32-WROOM-32D	
Geheugen	FLASH: 256 kB	SRAM: 32 kB	FLASH: 4 MB	SRAM: 520 kB
Interface	SPI/ I2C/ UART		SPI/ I2C/ UART	
Wifi	Ja		Ja	
BLE	Neen		Ja	
Voltage	5 V-input	3,3 V-output	5 V-input	3,3 V-output
Pinout	Digitaal/ analoog/ PWM		Digitaal/ analoog/ PWM	
Afmetingen	61,5 mm x 25 mm		52 mm x 28 mm	
Prijs	€ 33,50		€ 9,50	

In de vergelijking valt onmiddellijk de kostprijs van de Arduino op. De rest van de eigenschappen is zo goed als gelijk. Bij de variant van Espressif is het geheugen beter én is er de mogelijkheid om voor die goedkopere prijs zelfs BLE te hebben.

Op basis van de beschikbare informatie is Espressif de beste optie voor MCU's. De ESP32 is een heel goed ondersteunde microcontroller om de nodige doeleinden waar te maken. Dit hoeft niet precies de ESP-WROOM-32 te zijn, maar kan ook een variant zijn binnen deze familie, aangezien de meeste eigenschappen worden gedeeld. Tabel 4

Met de meest gepaste chip gekozen, is het belangrijk om te weten welke sensoren compatibel zijn met deze MCU's.

Er wordt eerst gekeken hoe een ESP32 van spanning en van stroom wordt voorzien als er alleen met de chip wordt gewerkt [8]. Een afbeelding van de chip Figuur 3: ESP32-WROOM-32D chip zonder pcb te zien.

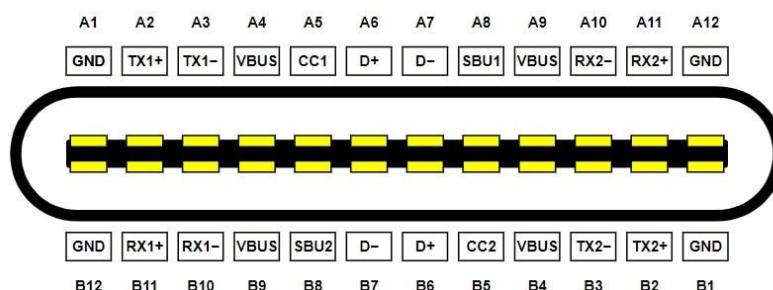


Figuur 3: ESP32-WROOM-32D chip zonder pcb [8]

1.2.1 Voeding

Volgens een bron van het Europees Parlement [9] wordt er vroeg of laat voor elk nieuw elektrisch apparaat op de handelsmarkt gebruik gemaakt van USB-C. Dit zou vanaf de herfstperiode van 2024 langzaam worden ingeburgerd.

Dit is een goede reden om USB-C te implementeren in het project. De connector heeft als eigenschap dat hij gebruiksvriendelijk is. Het maakt niet uit in welke richting de kabel wordt aangesloten, communicatie en voeding met het apparaat is er direct via de 24 pins-connector. Het grote voordeel van USB-C is dat er een hogere datatransmissiesnelheid is en een hogere stroomvoorziening van 100 W bij 20 V en 5 A. Een voorstelling van zo een connector Figuur 4.



Figuur 4: USB-C-pinout [9]

De voorstelling hierboven (Figuur 4), geeft de pinnen weer voor elk mogelijk protocol. Er zijn verschillende protocollen zoals Thunderbolt, Displayport, HDMI en USB die USB-C ondersteunen. Belangrijk voor de communicatie met de ESP-chip is het USB-protocol.

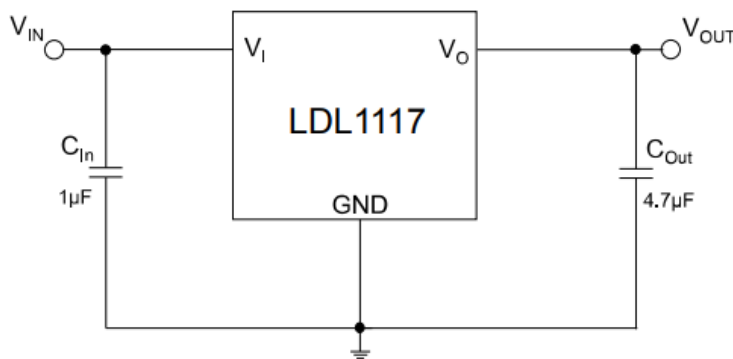
Om de ESP te voeden, is het belangrijk dat die de juiste spanning krijgt. De chip wordt gevoed met een spanning van 3,3 V en dit wordt met een low-drop-outregulator (LDO) verkregen.

Dit is een lineaire DC-spanningsregelaar die de uitgangsspanning stabiel kan regelen op 3,3 V.

Het systeem werkt volgens het principe met potentiaalverschil tussen input en output, de ingang moet altijd hoger zijn dan de uitgang. Het energieverlies tussen deze twee spanningen wordt omgezet in warmte.

Er is geen verklaring voor de specifieke waarde van de condensatoren die rond een LDO staan (Figuur 5). De elektrolytische condensatoren dienen in een schakeling als spanningsbuffer. Er is bijgevolg nog steeds een kleine spanning aanwezig als de algemene spanningsbron plots wegvalt in het circuit.

Keramische condensatoren zijn meestal heel laag van capaciteit en worden bij een LDO gebruikt als ontkoppelcondensatoren. Die dienen om ruis te filteren dat aanwezig is in een schakeling. [10]



Figuur 5: Typische applicatie LDO [10]

1.2.2 Communicatie

Om te communiceren met een ESP-32 moet er een aparte chip worden gebruikt. Eén van de veel gebruikte mogelijkheden is de CP2102. Uit de USB-C-poort komt een protocol dat USB heet, maar dit verstaat de ESP-32 niet. Wat de ESP wel verstaat is het UART-protocol. Dit is de reden waarom wordt gedacht aan de veel gebruikte CP2102, het is namelijk een USB naar TTL Serial UART-converter.

TTL staat voor Transistor-Transistor Logica en heeft een spanningsniveau van 0 tot 5 V. Het is een chip op basis van transistoren.

De CP2102 biedt een complete plug-and-play-interface-oplossing met royaltyvrije stuurprogramma's. De chip is USB 2.0-compatibel, wat zorgt voor snelle communicatie.

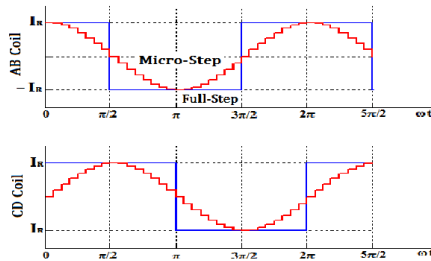
1.2.3 Programmatiecircuit

Er is een resetknop en een flashknop (of bootknop) vereist om een programma op de microcontroller te laden. De functie van de resetknop is om de enablepin van de MCU naar een ground te trekken omdat de enablepin standaard op een hoog niveau staat. Als dit gebeurt, dan wijkt de chip af van zijn standaardprogramma, wordt het onderbroken en ook gereset.

Om programma's in te laden, kan er gebruik worden gemaakt van een transistornetwerk om automatisch in te laden. Hiervoor is geen knop meer nodig. Als dit netwerk niet aanwezig is, wordt er tijdens het inladen van het programma op de flashknop gedrukt.

1.2.4 Sensor voor het aansturen van de stappenmotor

TMC2208 is een ultrastille tweefasige stappenmotordriverchip [11]. Het stuurt de motoren aan door full- en microstepping. TRINAMIC's geavanceerde stealthChop2 technologie zorgt voor een geruisloze werking, maximale efficiëntie en het beste motorkoppel. De snelle stroomregeling en de optionele combinatie met spreadCycle zorgen voor een zeer dynamische beweging.



Figuur 6: Full-step vs. micro-step [12]

De bedoeling van microstepping is het omzetten van de blokgolf die de motor aanstuurt naar een sinusgolf. Een perfecte sinusgolf wordt nooit behaald, maar de TMC2208 komt, in vergelijking met de A4988 en andere chips, er het dichtste bij omdat het gebruik maakt van 1/256-microstepping. Zo worden de stappen zeer klein en worden er minimale trillingen gegenereerd, waardoor deze chip de motor zeer stil doet draaien.

Zonder microstepping maakte bijvoorbeeld een 4-stappenmotor vier stappen per volledige aswenteling (dus 90° per stap). Door gebruik te maken van een 1/256-microstepping worden de motorspoelen zo aangestuurd dat er $4 * 256 = 1024$ stappen nodig zijn om een volledige omwenteling te maken (dus 0,3515625° per microstap).

Eigenschappen [13]:

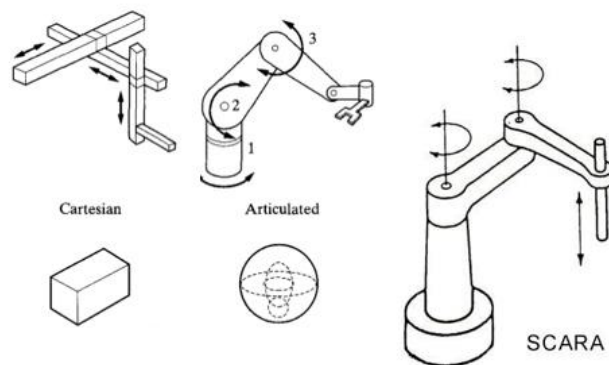
- Aandrijfcapaciteit tot 1,2 A (RMS) continue spoelstroom - 2 A-piek
- STEP/DIR-interface met 2, 4, 8, 16 of 32 microstep-pininstelling
- 256 microstappen door microPlyer™-interpolatie
- StealthChop2 voor stille werking en vloeiende bewegingen (standaardinstelling)
- SpreadCycle zeer dynamische motorbesturing chopper (schakelbaar via UART)
- Voedingsspanning motor: 5,5 tot 36 V
- Logische voedingsspanning: 3,3 tot 5 V
- Automatische stand-by stroomreductie (optie)
- Passief remmen en vrijloop
- Single Wire UART & OTP voor geavanceerde configuratie-instellingen
- Geïntegreerde pulsgenerator voor zelfstandige beweging

Een andere keuze om de motoren aan te sturen is de A4988 microstepping stappenmotordriver. Het voordeel van deze chip is dat hij een beetje goedkoper is. Het nadeel en de reden waarom deze chip niet wordt gebruikt, is dat de motoren veel meer lawaai maken.

1.3 Mechanica

In een vorig hoofdstuk werden de verschillende types motoren al besproken. Zoals werd onderzocht heeft ieder type motor een verschillende draagkracht, en is die dus ook beter toepasbaar op verschillende punten.

Bij industriële armen wordt voor de scharnierpunten gebruik gemaakt van stappenmotoren en servomotoren. Afhankelijk van het type motor en toepassing is een andere mechanische ontwerpstrategie nodig. Het prototype, besproken in dit dossier, wordt uitgewerkt met een 3D-printer. Ook dat is een belangrijk punt om rekening mee te houden. De verschillende mogelijkheden chassis en motormontage kunnen opgedeeld worden in drie categorieën. De groepen bestaan uit een cartesische, een articulated en een SCARA-robotarm, deze is te zien op de figuur hieronder. Figuur 7 [14]



Figuur 7: Verschillende types robotarmen [14]

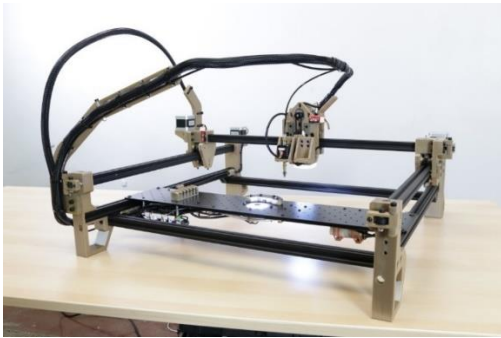
1.3.1 Cartesische robotarm



Figuur 8: Cartesische robotarm en de aluminium profielen [16] [15]

De cartesische robot lijkt het meeste op een klassieke CNC-machine of 3D-printer. Cartesische of lineaire robots werken op drie assen: X,Y en Z. Deze drie assen maken het mogelijk om in drie dimensies te bewegen: horizontaal (X- en Y-as) en verticaal (Z-as). Dit type robot is erg flexibel dankzij de specifieke opbouw. Zo kan die snel en eenvoudig worden aangepast, afhankelijk van de toepassing. Deze hoge flexibiliteit is te danken aan de typerende aluminiumprofielen waaruit de robot is opgebouwd. Die zijn internationaal verkrijgbaar in verschillende lengtes en breedtes en laten zich erg makkelijk bewerken. Hierdoor kunnen lagers, geleiders en andere componenten makkelijk en snel worden gemonteerd. Door deze standaardprofielen kan ook de prijs van een cartesisch systeem erg aantrekkelijk zijn. Figuur 8 [15] [16]

Een mogelijke toepassing van dit ontwerp is het open source LumenPnP-project door Opulo. Figuur 9 [17]



Figuur 9: LumenPnP cartesische robotarm [17]

Deze machine maakt het voor iedereen mogelijk printplaten op één niveau met hoge precisie te bestukken. De machine beweegt op de drie assen in rechte lijnen, een toepassing waar de cartesische robot uiterst geschikt voor is.

Het volledige werkgebied is echter wel afgebakend. Dit is een nadeel voor de robot die in dit onderzoek wordt gemaakt, aangezien bewegingsvrijheid een belangrijk punt is.

1.3.2 Articulated robotarm

Een tweede type is de articulated of vrijdraaiende robot. Dit type lijkt het best op een menselijke arm met alle bijbehorende voordelen. Deze robotarm wordt gemonteerd op een draaiende basis. Met een basisaantal motoren kan de volledige 3D-ruimte rondom de arm worden bereikt. Naarmate meer motoren en assen worden toegevoegd, wordt de toegankelijkheid van dit gebied nog verbeterd. Dit type robot kent een wijde inzetbaarheid in de industrie; van taken zoals lassen en materialen verplaatsen tot volledige assemblage en inpakken. Figuur 10/ 11 [18]



Figuur 10: Lasrobot[18]

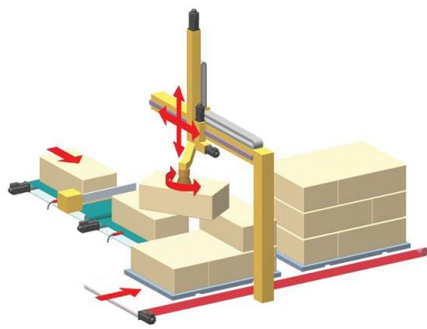


Figuur 11: Assemblagerobot [18]

Eén van de voordelen van dit ontwerp is de centrale basis waar de hele arm op draait, wat in dit onderzoek erg belangrijk is. Het volledige gewicht komt namelijk op één centrale motor en as terecht. De andere motoren werken wel mee aan de beweging, op voorwaarde dat de onderste as die kan ondersteunen.

Om dit allemaal mogelijk te maken, zijn overbrengingen op de motoren erg belangrijk.

1.3.3 SCARA-robotarm



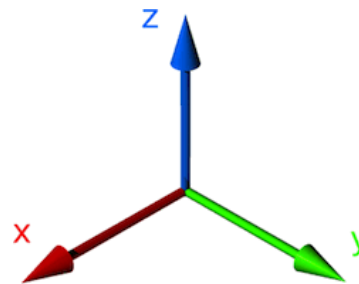
Figuur 12: Scara-robotarm [15]

Het SCARA-ontwerp heeft, net zoals de cartesische robot, drie assen waarop die beweegt. Het verschil tussen beide is de manier waarop wordt bewogen. Het cartesische ontwerp doet dat telkens in een hoek van 90° , terwijl een SCARA-robot om zijn eigen as kan draaien. Dat maakt dit ontwerp uiterst geschikt voor het stapelen van dozen of assemblage. Figuur 12 [15]

Voor dit ontwerp zijn verschillende opensourcetoepassingen beschikbaar. Bijvoorbeeld dit type door PyBot. [19]



Figuur 13: Uitgewerkt versie PyBot [19]



Figuur 14: 3D-Assenstelsel [19]

Het is opnieuw volledig uitgewerkt op een 3D-printer met standaardcomponenten erin verwerkt. Dit type geeft door zijn enkele roterende basis een beter beeld van de toepassing in dit onderzoek. Hij is gelimiteerd in de bewegingen die worden gemaakt. Zo is iedere beweging in de Z-richting of hoogte afhankelijk van één motor. Figuur 13/ 14

Vanwege de vrije bewegingen in een openruimte en de vele informatie die erover beschikbaar is, wordt in dit onderzoek voor een articulating arm gekozen.

1.4 Veiligheid

Bij het ontwerpen en realiseren van een robotarm zijn verschillende aspecten van veiligheid van belang. Het uitvoeren van een goede risicoanalyse is cruciaal om mogelijke gevaren te identificeren. Deze gevaren zouden zich kunnen voordoen tijdens het gebruik van de robotarm. Het is daarbij van belang om potentiële gevaren voor zowel de bediener als andere personen in de omgeving van de robotarm te evalueren.

Stap één van de risicoanalyse omvat de identificatie van mogelijke gevaren, zoals snelle bewegingen van de robotarm die letsels kunnen veroorzaken, hoge temperaturen bij het gebruik van de robotarm, blootstelling aan elektrische stroom en mogelijke botsingen met personen of objecten in de buurt.

Stap twee van de risicoanalyse omvat de evaluatie van de waarschijnlijkheid dat de gevaren zich zullen voordoen en de mogelijke ernst van de gevolgen als ze zich voordoen. Op basis hiervan kan worden bepaald welke risico's prioriteit hebben bij het nemen van beheersmaatregelen.

Stap drie van de risicoanalyse omvat het identificeren van beheersmaatregelen om de risico's te verminderen of te elimineren. Dit kan bijvoorbeeld worden bereikt door het installeren van fysieke barrières rondom de robotarm of door gebruik te maken van sensoren die de robotarm kunnen stoppen wanneer een persoon in de buurt komt. Het is daarbij van belang om noodstoppen te installeren die de robotarm onmiddellijk stoppen wanneer er een onverwachte situatie ontstaat. Verder moeten alle gebruikers van de robotarm getraind zijn om veilig gebruik te maken van de robotarm en eveneens van de bijhorende apparatuur. Ook regelmatig onderhoud en inspectie zijn belangrijk om preventief te kunnen handelen.

Door rekening te houden met deze factoren en andere belangrijke veiligheidsaspecten, kunnen ontwerpers en gebruikers ervoor zorgen dat de robotarm veilig en efficiënt werkt.
[20]

1.5 Opzetstuk

Eén van de belangrijkste aspecten bij het maken van een robotarm is het ontwerpen van de opzetstukken. Een opzetstuk is het deel van de robotarm dat in contact komt met het object dat wordt verplaatst of bewerkt. Het type opzetstuk dat wordt gebruikt, hangt af van de toepassing en het gewenste resultaat.

Eén van de meest voorkomende opzetstukken voor een robotarm is de gripper. Dit opzetstuk wordt gebruikt om objecten vast te pakken en te verplaatsen. Er zijn verschillende soorten grippers, zoals parallelle grippers, hoekgrippers en pneumatische grippers. Parallelle grippers worden vaak gebruikt bij industriële toepassingen, zoals het assembleren van producten, het verpakken van goederen of het laden en lossen van vrachtwagens. Ze zijn ook populair in laboratoriumomgevingen en in de gezondheidszorg voor het hanteren van delicate en gevoelige objecten. Over het algemeen bieden parallelle grippers een betrouwbare en efficiënte manier om objecten vast te pakken en te hanteren met behulp van een robotarm. Figuur 15



Figuur 15: Parallelle gripper [36]

Hoekgrippers zijn een ander type gripper dat vaak wordt gebruikt. Zoals de naam al aangeeft, zijn hoekgrippers ontworpen om objecten vast te pakken en vast te houden met behulp van een set vingers die in een hoek zijn gepositioneerd ten opzichte van elkaar. Dit type gripper wordt gebruikt in toepassingen waarbij het vast te pakken voorwerp niet plat of rechthoekig is, maar een onregelmatige vorm heeft. Pneumatische grippers worden voornamelijk gebruikt voor het hanteren van delicate objecten zoals objecten die gemaakt zijn van glas of breekbaar materiaal. Figuur 16



Figuur 16: Pneumatische hoek-gripper [37]

Een ander type opzetstuk is een lasersnijder. Dit opzetstuk gebruikt lasers om materialen te snijden. Het is vooral handig bij industriële toepassingen waar precisie en snelheid van belang zijn, zoals bij het snijden van metaalplaten.

Schuurmachines zijn een andere soort veelvoorkomende types van opzetstukken voor een robotarm. Ze worden gebruikt om materialen te schuren en glad te maken. Dit type opzetstuk wordt het meest gebruikt in de automobiel- en meubelindustrie.

Een magnetisch opzetstuk wordt gebruikt om metalen voorwerpen op te tillen en te verplaatsen. Dit type opzetstuk is handig voor het hanteren van zware of gevaarlijke metalen voorwerpen. Dit kan bijvoorbeeld van pas komen in een fabriek waar zware metalen onderdelen worden geproduceerd. Het gebruik van een magnetisch opzetstuk kan de efficiëntie en productiviteit van de robotarm verbeteren. Het kan immers snel en gemakkelijk objecten oppakken en verplaatsen zonder dat er veel aanpassingen nodig zijn. Bovendien verkleint de kans op ongevallen door vallende voorwerpen omdat ze beter vasthangen aan de robotarm.

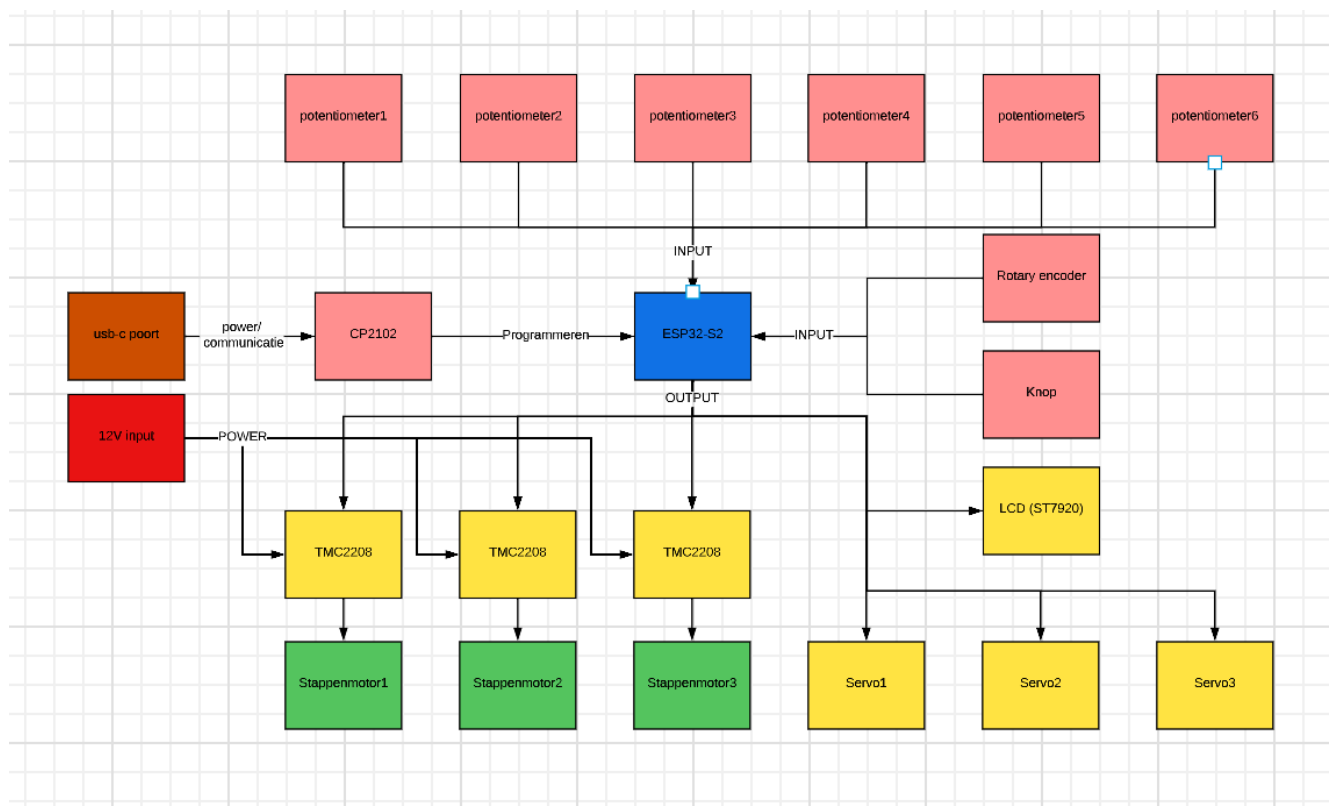
Het is ook mogelijk om camerasystemen te gebruiken als opzetstuk. Dit type opzetstuk wordt gebruikt om beelden vast te leggen en te analyseren. Het wordt vaak gebruikt in de industrie voor kwaliteitscontrole, inspectie en robotica-onderzoek. Het camerasysteem kan er in combinatie met andere types opzetstukken voor zorgen dat bijvoorbeeld delicate voorwerpen beter gegrepen kunnen worden omdat er voorziening is van een beter beeld.

Men kan een robotarm ook uitrusten met een boormachine als opzetstuk om gaten te maken in materialen. Dit soort opzetstuk is ideaal voor situaties waarbij precisie, consciëntie en herhaalbaarheid van cruciaal belang zijn. Een robotarm met een boormachineopzetstuk kan ook worden gebruikt om schroeven te draaien in onderdelen die al voorzien zijn van schroefgaten. Dit vermindert de noodzaak van menselijke arbeid en verbetert de efficiëntie van het productieproces. Naast boren en schroeven kan de robotarm gebruik maken van een boormachineopzetstuk om een bepaald patroon in een materiaal te frezen.

De keuze in gebruik van het soort opzetstuk is belangrijk in het designproces van de robotarm. Hij kan op die manier volledig naar wens worden gemaakt en er kan eventueel extra gewicht worden toegevoegd aan de basis van de arm zodat hij objecten van 250 tot 500 gram kan tillen.

2 Ontwerp

2.1 Blokschema



Figuur 17: Blokschema

Dit blokschema zorgt ervoor dat er een duidelijk overzicht is van alle belangrijke componenten die worden gebruikt in het project. Wat opvalt, is dat er verbindingsen worden gemaakt tussen de blokken. De pijlen geven de richting aan waarin de communicatie loopt.

De functie van de pijlen – input/ output, programmeren, power – staat erbij vermeld.

De ESP32-S2 is hier duidelijk het centrale communicatiepunt. Dit is logisch omdat deze chip het brein van het systeem vormt. De chip stuurt signalen zoals PWM/ I2C/ SPI uit via de outputs en kan analoge ingangen uitlezen van bijvoorbeeld een potentiometer.

Het blokschema heeft een kleurencode, hierdoor wordt in één oogopslag duidelijk hoe de werking en opstelling van het bord is.

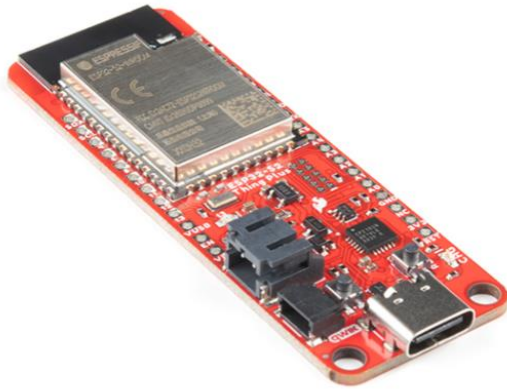
De legende voor figuur 17 is als volgt:

- Roze: alle inputsystemen zoals de communicatie voor de chip en de analoge ingangen
- Geel: alle outputs naar drivers/ servomotoren en een lcd
- Groen: de stappenmotoren die afhankelijk zijn van een driver zoals de TMC2208
- Blauw: het brein van het systeem, de ESP32-S2
- Rood: voeding naar het systeem

Het blokschema is nuttig aangezien het een vertrekpunt is voor een elektrisch schema. Het elektrische schema wordt besproken in hoofdstuk 2.5.

2.2 Componenten

Alle componenten kunnen individueel worden gerealiseerd. Daarom wordt elk belangrijk onderdeel van het schema eerst apart besproken. In dit geval wordt de ESP32-S2 als eerste besproken.



Figuur 18: SparkFun Thing Plus - ESP32-S2 WROOM [21]



Figuur 19: ESP32-S2 chip [22]

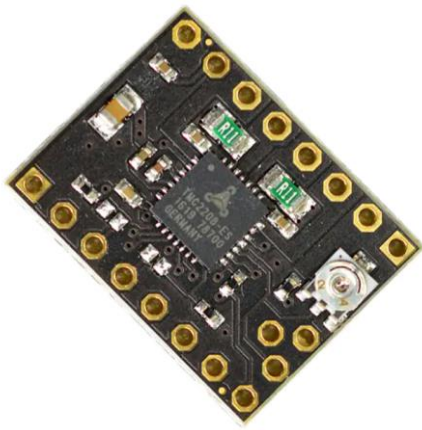
Om van start te gaan met de ESP32-S2 is het belangrijk de eigenschappen te bekijken van de chip. Enkele belangrijke eigenschappen zijn de aanwezigheid van USB-C, wifi en een groter aantal GPIO-pinnen dan een standaard ESP32.

Het gebruik van de ESP32-S2 is ideaal voor dit project, aangezien de bluetooth-functionaliteit van de chip is vervangen door extra GPIO-pinnen. Dit betekent dat deze microcontroller beschikt over 43 GPIO-pinnen die kunnen worden geconfigureerd voor verschillende functies.. [22]

Het Sparkfunboard zorgt voor een goede start en basisdocumentatie om een schema op te bouwen. [21]

Bij deze microcontroller is het vooral belangrijk dat de wifiantenne volledig vrij blijft en geen interferentie ondervindt van draden of sporen op de pcb. Daarnaast heeft de ESP een ingebouwde standaard USB-verbinding die vrij moet blijven om externe connectiviteit mogelijk te maken. Figuur 18/ 19

2.2.1 TMC2208



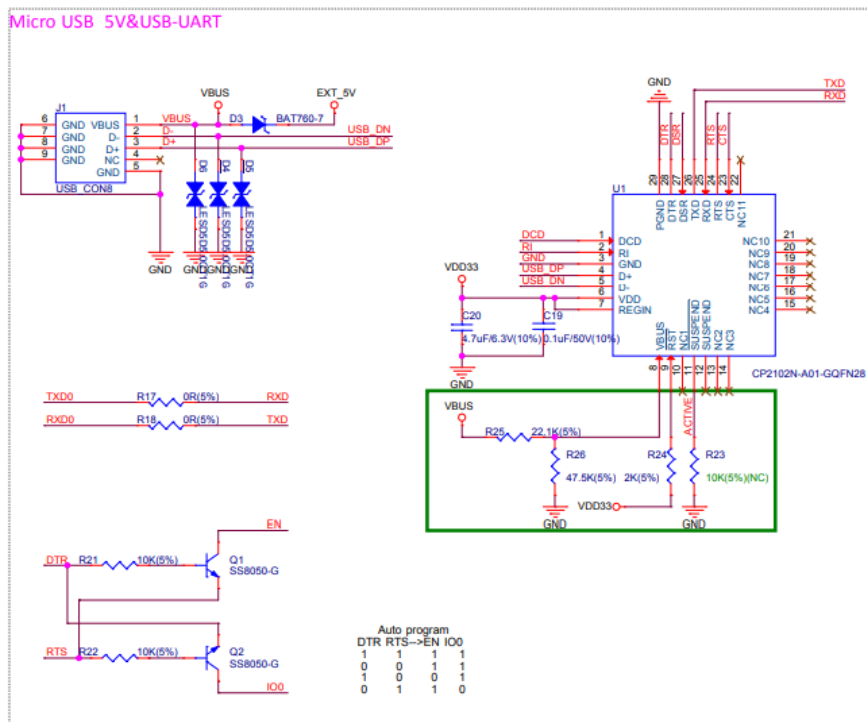
Figuur 20: Digikey TMC2208 board [23]

Bij het ontwerpen van de pcb is rekening gehouden met de stroom die de TMC moet krijgen. Aangezien deze driver een motor aanstuurt, levert de TMC een maximale stroom van 1,2 A en daarom is de baandikte bij de pcb berekend zoals in hoofdstuk 2.6 wordt besproken. De stroom is instelbaar met de potentiometer die aanwezig is per driver op de pcb. Op de foto is de potentiometer te zien als het metalen object dat zich uiterst rechts op de pcb bevindt.

Op zijn website stelt de fabrikant veel informatie ter beschikking. [23] Digikey heeft veel ervaring bij het ontwikkelen van een prototype van deze module. Momenteel bestaat versie twintig van het elektrische schema en daarom is er gebruik van gemaakt. Figuur 20

Bij het implementeren op een eigen printplaat moet rekening worden gehouden met enkele factoren. Aangezien de chip heel warm kan worden vanwege het stroomverbruik, is het raadzaam om aandacht te besteden aan een goede koeling, zoals het gebruik van een koelvin indien nodig.

2.2.2 CP2102



Figuur 21: CP2102 schema [24]

De CP2102 is een veel gebruikte chip om te communiceren van USB naar UART. Door gebruik te maken van deze chip, kan er seriële communicatie worden gevormd. Het debuggen wordt op deze manier gemakkelijker. Bij de meeste standaardmodules van een ESP32 worden deze chips gebruikt.

Het standaardschema is open source gezet door de fabrikant, wat een goede gelegenheid biedt om de CP2102-chip te verwerken met het Sparkfunschema. Het programmatienetwerk is hieraan verbonden. Figuur 21 [24]

2.2.3 ST7920



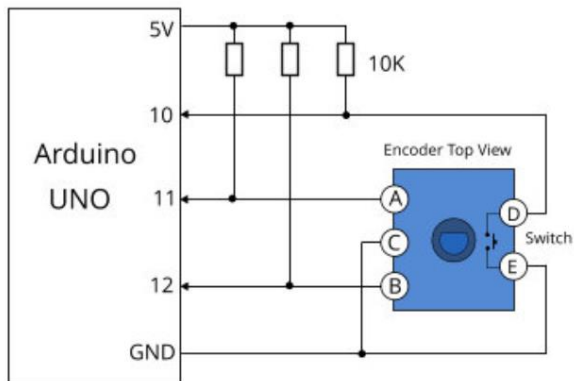
Figuur 22: ST7920 lcd [25]

De ESP32-S2 heeft meerdere SPI-poorten, en de ST7920-LCD maakt gebruik van dit protocol om met de ESP32-S2 te communiceren. [25] Het is interessant om te weten welke pinnen van de chip daarvoor in gebruik worden gesteld zonder extra instellingen te moeten doen. Er is namelijk maar één poort die wordt verbonden en rechtstreeks kan worden geprogrammeerd zonder aanpassingen te moeten doen of zonder extra commando's te schrijven naar de ESP. Figuur 22 Figuur 23 [26]

Device	Signal	Pin	Symbol	Remarks
SPI_HOST0 / SPI_HOST1	SPICS0	GPIO29	-	reserved for flash and PSRAM
SPI_HOST0 / SPI_HOST1	SPICS1	GPIO26	-	reserved for flash and PSRAM
SPI_HOST0 / SPI_HOST1	SPICLK	GPIO30	-	reserved for flash and PSRAM
SPI_HOST0 / SPI_HOST1	SPID	GPIO32	-	reserved for flash and PSRAM
SPI_HOST0 / SPI_HOST1	SPIQ	GPIO31	-	reserved for flash and PSRAM
SPI_HOST0 / SPI_HOST1	SPIHD	GPIO27	-	reserved for flash and PSRAM (only in qio or qout mode)
SPI_HOST0 / SPI_HOST1	SPIWP	GPIO28	-	reserved for flash and PSRAM (only in qio or qout mode)
SPI_HOST0 / SPI_HOST1	SPIIO4	GPIO33	-	reserved for Flash and PSRAM (only in octal mode)
SPI_HOST0 / SPI_HOST1	SPIIO5	GPIO34	-	reserved for Flash and PSRAM (only in octal mode)
SPI_HOST0 / SPI_HOST1	SPIIO6	GPIO35	-	reserved for Flash and PSRAM (only in octal mode)
SPI_HOST0 / SPI_HOST1	SPIIO7	GPIO36	-	reserved for Flash and PSRAM (only in octal mode)
SPI_HOST0 / SPI_HOST1	SPIDQA	GPIO37	-	reserved for Flash and PSRAM (only in octal mode)
SPI_HOST2 (FSPI)	SCK	GPIO12	#SPI0_SCK	can be used
SPI_HOST2 (FSPI)	MOSI	GPIO11	#SPI0_MOSI	can be used
SPI_HOST2 (FSPI)	MISO	GPIO13	#SPI0_MISO	can be used
SPI_HOST2 (FSPI)	CS0	GPIO10	#SPI0_CS0	can be used

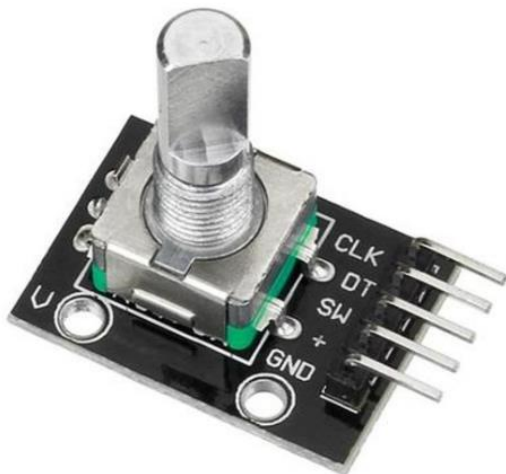
Figuur 23: SPI pinnen [26]

2.2.4 Rotary encoder



Figuur 24: Rotary encoder schema [27]

De rotary aansturen is vrij eenvoudig: enkele GPIO-pinnen zijn voldoende om alle data in te lezen. Om een rotary encoder aan te sturen zonder dat die op een pcb gemonteerd is, zoals op de onderstaande foto, moeten nog enkele weerstanden worden toegevoegd. Figuur 24
Figuur 25 [27] [28]

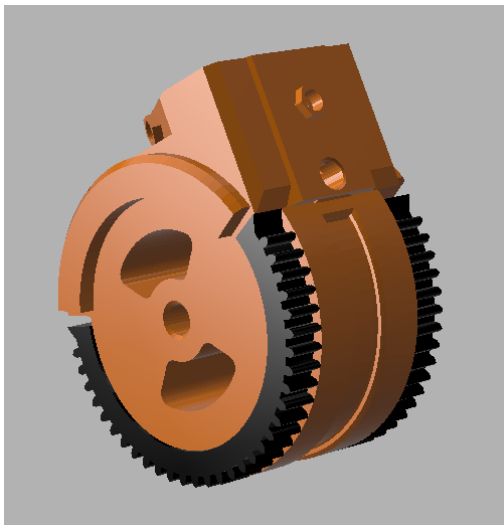


Figuur 25: Rotary encoder op pcb [28]

2.3 Mechanisch ontwerp

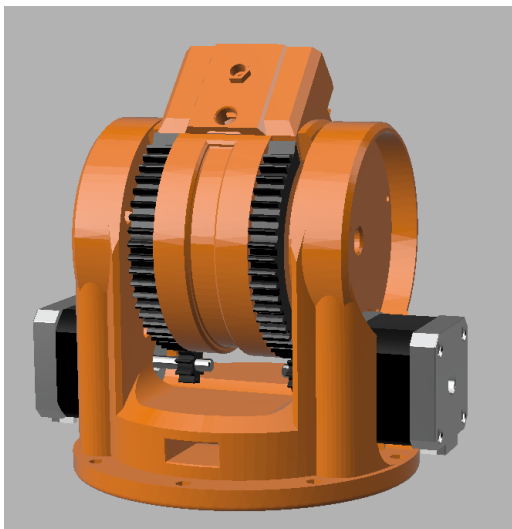
In de onderzoeksfase werd besloten dat de robotarm gebruik maakt van stappenmotoren voor de aandrijving van de assen die de grootste last dragen. De bewegingen worden daarom via een tandwieloverbrenging doorgevoerd. Die kan worden uitgewerkt op een 3D-printer waardoor een volledig specifieke vertanding wordt ontworpen. De overbrenging bestaat uit twee delen: het stationaire deel op de as en het draaiende deel aan de motor.

De vertanding rechtstreeks aanbrengen op de onderdelen geeft een extra uitdaging voor het printen omdat de printer altijd een vlak nodig heeft om op te werken. Om die reden wordt het geheel in twee kleinere stukken onderverdeeld. [29]



De twee delen zijn zichtbaar door het kleurverschil in Figuur 26. Het grotere oranje deel wordt op een lagere resolutie geprint om de printtijd te verkorten. De vertanding wordt op een hogere resolutie geprint om de kwaliteit te verhogen. Deze twee onderdelen worden dan samen verlijmd tot één geheel.

Figuur 26: Uitwerking met geprinte vertanding



Vervolgens kan de statische vertanding worden verbonden met de motoren door een kleiner tandwiel. Dit tandwiel heeft een groef waardoor de draaibeweging wordt overgebracht. Figuur 27

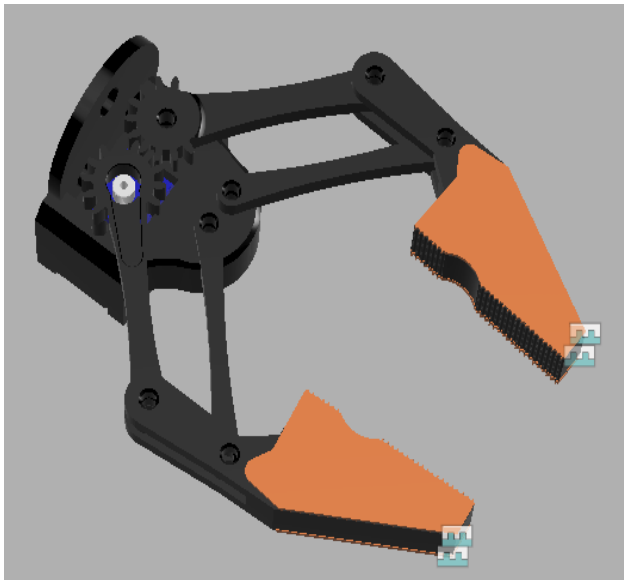
Figuur 27: Uitwerking met motoren

Een andere beslissing die werd gemaakt, is om de verschillende assen door te verbinden met buizen. Zo wordt printtijd bespaard en de sterkte van het geheel verbeterd. Zoals in Figuur 28 zichtbaar is, kunnen alle assen worden onderverdeeld om de belastende krachten te verdelen. De buizen hebben een diameter van 40 en 35 mm.



Figuur 28: Connectie tussen de assen met buizen

In de onderzoeksfase werden verschillende opties overlopen voor de grijper. Die zal in dit prototype bestaan uit twee kleine “vingers” die via een servomotor kunnen sluiten.



Figuur 29: Grijper

Zoals zichtbaar is in Figuur 29 wordt de draaiende beweging van de servomotor omgezet in een beweging die de “vingers” laat sluiten.

Dit gebeurt via een tandwieloverbrenging die met één servomotor beide “vingers” bestuurd. Door een extra stuurarm toe te voegen, blijven de “vingers” ook steeds op een symmetrische afstand van elkaar.

Via een servomotor, die op een vorige as wordt gemonteerd, is deze grijper in staat rond de centrale as te draaien. Dit maakt het optillen en het omdraaien van voorwerpen mogelijk.

2.4 BOM (Bill of Materials)

De BOM is een stukkenlijst die is opgesteld voor elke gebruikte component.

Een stukkenlijst is van groot belang om een idee te krijgen van het kostenplaatje van het project. Doordat de link elke keer vermeld staat bij het product, is het praktisch om alles makkelijk terug te vinden en daarom is de prijs per stuk en de algemene prijs weergegeven. Een groot deel van de componenten is aangekocht bij AliExpress omdat dit goedkoper is.

Om de gebruikte BOM te bekijken, wordt verwezen naar Bijlage 4

2.5 Schema

Het schema is te vinden in Bijlage 5.

Voor de realisatie van het schema wordt het programma Eagle gebruikt. Met de software van het bedrijf Autodesk is het mogelijk om een elektronisch schema te maken dat naar een boarddesign kan worden omgezet. Autodesk biedt de service Fusion 360 aan die het toestaat om een board design te converteren naar een 3D-ontwerp, wat het tekenen van een behuizing vereenvoudigt.

Om het elektronische schema geordend te houden, wordt het blad opgedeeld door stippellijnen. Elk vak bevat de naam van het bijbehorende circuit. Een connectie kan worden gemaakt met een draad of met labels, alle labels met dezelfde naam zijn doorverbonden. Eveneens zijn alle gebruikte pinnen extern beschikbaar om metingen te doen of om apparatuur op aan te sluiten.

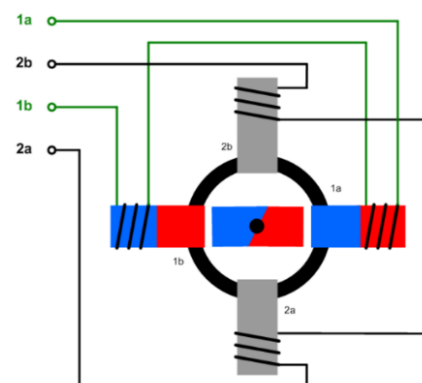
De bedoeling is dat de USB-C-connector dient om de microcontroller te programmeren. Dit is een standaard aan het worden zoals aangekaart bij hoofdstuk 1.2.1. Het is gemakkelijk in gebruik, aangezien het niet uitmaakt hoe de kabel in de poort wordt gestoken, het werkt altijd.

Om het circuit te programmeren, wordt gebruik gemaakt van twee knoppen die dienen om een programma op de chip te plaatsen. De ene pin is een bootpin en de andere pin is een resetpin. Deze knoppen worden niet gebruikt, aangezien dit proces door twee transistoren kan worden geautomatiseerd. Als back-up zijn er wel nog twee knoppen aanwezig.

Enkele belangrijke componenten zijn elco's en keramische condensatoren. Lage capaciteiten tussen pico- en microfarad zorgen ervoor dat ruis weg te filteren, de spanning te stabiliseren en DC-signalen te blokkeren.

Hoge capaciteiten (bereik van enkele farads) zorgen ervoor dat er bepaalde lading kan worden opgeslagen.

Er is ook een sense-weerstand aanwezig in het project met een weerstand van 0,11 ohm. Er wordt gebruik gemaakt van twee weerstanden, één voor spoel A in de motor en één voor spoel B in de motor. Op die manier kan de stroom in de spoel worden gemeten. Figuur 30 [30]



Figuur 30: Spoelen in een stappenmotor [29]

2.6 Board design

De printplaat is te vinden in Bijlage 6.

Door het grote aantal componenten en de plaatsing van de pcb wordt er een 4-lagen pcb ontworpen. Een 4-lagen pcb is een printplaat met vier lagen koperen geleiders en wordt vaak gebruikt voor complexe elektronica. Bij het ontwerpproces zijn er verschillende belangrijke factoren om rekening mee te houden.

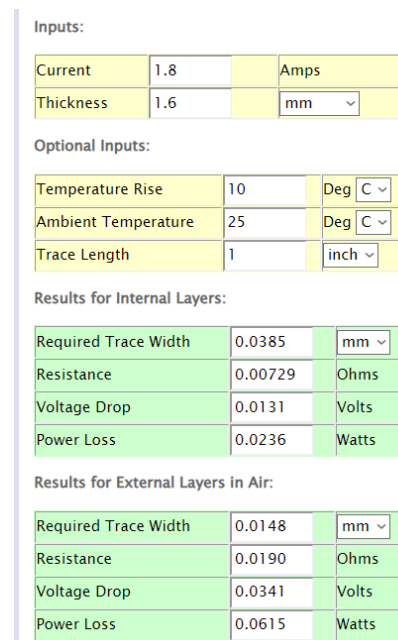
Allereerst is de lay-out van het pcb van groot belang. Het ontwerp moet zorgvuldig worden gepland om ervoor te zorgen dat alle onderdelen op de juiste manier kunnen worden geplaatst en verbonden. De plaatsing en routing van de koperbanen kan van groot belang zijn voor de prestaties van de pcb. In dit project wordt gebruik gemaakt van een lcd van 93 mm x 70 mm om op de pcb te plaatsen. Dit betekent dan de componenten met koelelementen zorgvuldig moesten worden gepositioneerd om te voorkomen dat ze in de weg staan van de lcd.

Een andere belangrijke factor is thermische beheersing. Op de printplaat staan enkele componenten die warmte genereren, dus moet ervoor worden gezorgd dat de warmteafvoer van deze componenten goed verloopt en dat ze geen problemen veroorzaken. Dit kan worden bereikt door het gebruik van koperbanen en thermische pads, die op hun beurt de warmteafvoer verbeteren en voorkomen dat de componenten oververhit raken.

Daarnaast is stroomvoorziening een cruciale factor bij het ontwerpen van een pcb. Tijdens het ontwerp werd er rekening mee gehouden dat de voedingsbanen op de pcb dik genoeg zijn zodat deze banen niet doorbranden als er te veel stroom door loopt. Dit kan worden bereikt door het gebruik van brede koperbanen. Deze koperbanen worden berekend door gebruik te maken van een pcb-trace width calculator.

Voor de correcte specificatie dienen enkele gegevens te worden ingevoerd, zoals de maximale stroomcapaciteit van de koperbaan en de dikte van de pcb. Na het invullen van de gegevens, is de vaststelling dat er een baandikte moet worden voorzien van 0,0385 mm.

Figuur 31 [31]



Inputs:

Current	1.8	Amps
Thickness	1.6	mm

Optional Inputs:

Temperature Rise	10	Deg C
Ambient Temperature	25	Deg C
Trace Length	1	inch

Results for Internal Layers:

Required Trace Width	0.0385	mm
Resistance	0.00729	Ohms
Voltage Drop	0.0131	Volts
Power Loss	0.0236	Watts

Results for External Layers in Air:

Required Trace Width	0.0148	mm
Resistance	0.0190	Ohms
Voltage Drop	0.0341	Volts
Power Loss	0.0615	Watts

Figuur 31: Calculator voor baandikte [31]

3 Realisatie

3.1 Bestukking

Het bestukkingsproces is een cruciale stap bij het fabriceren van de printplaat, waarbij de elektronische componenten op de pcb worden geplaatst en gesoldeerd. In het bestukkingsproces werd gebruik gemaakt van een soldeeroven en een pcb-stencil, twee hulpmiddelen die efficiëntie en nauwkeurigheid bieden bij het plaatsen van componenten.

Bij dit proces speelt de soldeeroven, ook wel bekend als een reflow-oven, een belangrijke rol. De soldeeroven verwarmt de pcb tot een specifieke temperatuur, waarbij de soldeerpasta smelt en de componenten stevig op hun plaats worden gesoldeerd. Hierdoor ontstaat een sterke verbinding tussen de componenten en de pcb.

Voordat de pcb in de soldeeroven wordt geplaatst, is het noodzakelijk om soldeerpasta aan te brengen op de juiste plaatsen. Soldeerpasta is een mengsel van soldeertin en flux. Deze wordt aangebracht op de contactpads van de pcb waar vervolgens de componenten op geplaatst worden.

Om ervoor te zorgen dat de soldeerpasta nauwkeurig wordt aangebracht, wordt een pcb stencil gebruikt. Een pcb-stencil is een metalen sjabloon dat precies is uitgesneden om de soldeerpasta op de juiste plaatsen op de pcb aan te brengen. (Figuur 32) De stencil wordt bovenop de pcb geplaatst en de soldeerpasta wordt er doorheen aangebracht door hulp van een rakel of spatel. Dit zorgt voor een consistente en nauwkeurige aanbrenging van de soldeerpasta.

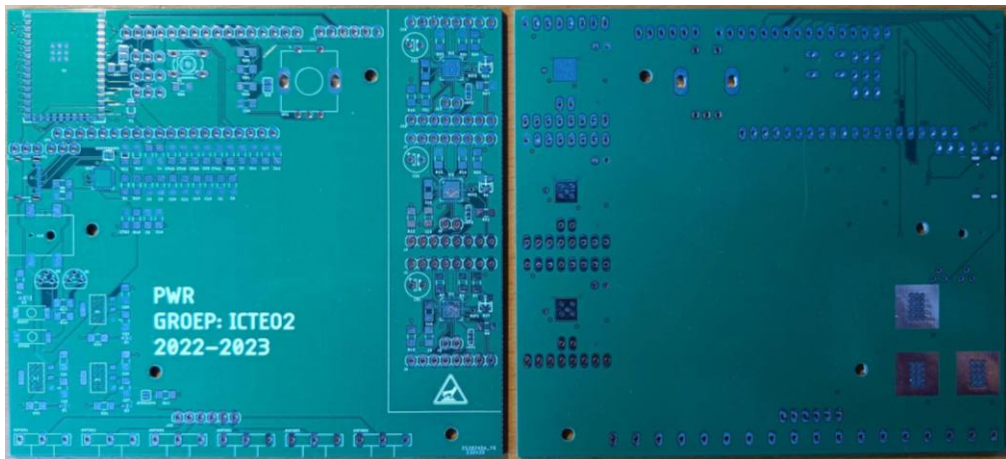


Figuur 32: Stencil voor de pcb

Het bestukkingsproces is samen te vatten in volgende stappen:

1. Voorbereiding: het schoonmaken van de pcb.
2. Plaatsing van de pcb-stencil: het positioneren van de pcb-stencil zodat de uitsparingen in de stencil overeenkomen met de contactpads op de pcb.
3. Aanbrengen van soldeerpasta: gebruik een rakel of spatel om de soldeerpasta gelijkmatig over de stencil te verdelen en de openingen te vullen.
4. Verwijderen van de stencil: voorzichtig de stencil verwijderen van de pcb en controleren of de soldeerpasta correct is aangebracht op de contactpads.
5. Plaatsing van de componenten: nauwkeurig plaatsen van alle elektronische componenten op de juiste posities op de pcb, waarbij de componenten in de soldeerpasta worden geplaatst.

Vervolgens kan de pcb in de soldeeroven, waar de soldeerpasta smelt en de componenten stevig op hun plaats worden gesoldeerd. Figuur 33



Figuur 33: Pcb met soldeerpasta



Figuur 34: Werkend resultaat van de gesoldeerde pcb

Het gebruik van een soldeeroven en een pcb-stencil vereenvoudigt het bestukingsproces en zorgt voor consistente resultaten. Deze benadering biedt efficiëntie en precisie, wat van essentieel belang is bij de productie van elektronische apparaten. Figuur 34

3.2 Programmatie

Bij het programmeren wordt er gestart met het toevoegen van de nodige bibliotheken in Arduino (Codefragment 1) . [32] [33] [34]

```
#include "AccelStepper.h"
#include <Servo.h>
```

Codefragment 1: Toevoegen bibliotheken [32] [33]

Initialisatie van de stappenmotoren en de servomotoren (Codefragment 2). [32] [33]

```
// AccelStepper Setup
// 1 = Easy Driver interface
//Arduino Uno Pin 3 connected to STEP pin of Easy Driver
//Arduino Uno Pin 4 connected to DIR pin of Easy Driver
AccelStepper stepper(1, 3, 4);
AccelStepper stepper2(1, 5, 6);
AccelStepper stepper3(1, 7, 8);

Servo servo1;
Servo servo2;
Servo servo3;
```

Codefragment 2: Initialisatie stappenmotoren en servomotoren [32] [33]

In de onderstaande code worden de knoppen gedefinieerd en worden er variabelen aangemaakt om de posities van de knoppen bij te houden (Codefragment 3). Er wordt een variabele buttonPushCounter gemaakt om het aantal keer dat er gedrukt wordt op te slaan.

```
// Variables to store current, previous and move position
//define the buttons
const int buttonPin1 = 12; // the pin that the pushbutton is attached to
int buttonState1 = 0;      // current state of the up button
int lastButtonState1 = 0;  // previous state of the up button

const int buttonPin2 = 13; // the pin that the pushbutton is attached to
int buttonState2 = 0;      // current state of the up button
int lastButtonState2 = 0;  // previous state of the up button
boolean button2Pressed = false; //define variable for values of the button

const int buttonPin3 = 2;  // the pin that the pushbutton is attached to

int buttonPushCounter;
```

Codefragment 3: Definiëren knoppen en variabelen aanmaken

Definiëren van de leds en de potentiometers. Er worden variabelen aangemaakt om de waarde van de potentiometers op te slaan en de hoek van de potentiometer te bepalen (Codefragment 4).

```
//define leds
#define led1 1
#define led2 0

//define potentiometers
const int pot1 = A0;
const int pot2 = A1;
const int pot3 = A2;
const int pot4 = A3;
const int pot5 = A4;
const int pot6 = A5;

//define variable for values of the potentiometers
int pot1Val;
int pot2Val;
int pot3Val;
int pot4Val;
int pot5Val;
int pot6Val;

//define variable for angles of the potentiometer
int pot1Angle;
int pot2Angle;
int pot3Angle;
int pot4Angle;
int pot5Angle;
int pot6Angle;
```

Codefragment 4: Definiëren led en potentiometers en variabelen voor de potentiometers

Definiëren van variabelen om de posities van de servomotoren op te slaan (Codefragment 5).

```
//define variable for saved position of the servos
int stepper1PosSave[] = { 1, 1, 1, 1, 1, 1, 1, 1 };
int stepper2PosSave[] = { 1, 1, 1, 1, 1, 1, 1, 1 };
int stepper3PosSave[] = { 1, 1, 1, 1, 1, 1, 1, 1 };
int servo4PosSave[] = { 1, 1, 1, 1, 1, 1, 1, 1 };
int servo5PosSave[] = { 1, 1, 1, 1, 1, 1, 1, 1 };
int servo6PosSave[] = { 1, 1, 1, 1, 1, 1, 1, 1 };
```

Codefragment 5: Definiëren variabelen voor posities

In de setup worden de snelheden van de stappenmotoren ingesteld. De servomotoren worden toegewezen aan een pin. De knoppen worden ingesteld als een input en er wordt één pin ingesteld als interruptie zodat waar ook in de code deze altijd wordt uitgevoerd. De leds worden ingesteld als een output (Codefragment 6). [32] [33]

```
void setup() {

    stepper.setMaxSpeed(4800);          // Set speed fast enough to follow pot
rotation
    stepper.setAcceleration(4800);      // High Acceleration to follow pot rotation
    stepper2.setMaxSpeed(4800);        // Set speed fast enough to follow pot
rotation
    stepper2.setAcceleration(4800);    // High Acceleration to follow pot rotation
    stepper3.setMaxSpeed(4800);        // Set speed fast enough to follow pot
rotation
    stepper3.setAcceleration(4800);    // High Acceleration to follow pot rotation

    servo1.attach(9);
    servo2.attach(10);
    servo3.attach(11);
    //define buttons as input units
    pinMode(buttonPin1, INPUT);
    pinMode(buttonPin2, INPUT);
    pinMode(buttonPin3, INPUT);
    attachInterrupt(digitalPinToInterrupt(buttonPin3), buttonISR, CHANGE);
    pinMode(led1, OUTPUT);
    pinMode(led2, OUTPUT);
}
```

Codefragment 6: Definiëren variabelen voor posities [32] [33]

In het begin van de loop worden de potentiometers uitgelezen en opgeslagen. Deze opgeslagen waarden worden dan gebruikt om de stappenmotoren en servomotoren in positie te brengen (Codefragment 7). [32] [33]

```
void loop() {

    pot1Val = analogRead(pot1);
    pot1Angle = map(pot1Val, 0, 1023, 0, 1600);
    pot2Val = analogRead(pot2);
    pot2Angle = map(pot2Val, 0, 1023, 0, 1600);
    pot3Val = analogRead(pot3);
    pot3Angle = map(pot3Val, 0, 1023, 0, 1600);
    pot4Val = analogRead(pot4);
    pot4Angle = map(pot4Val, 0, 1023, 0, 180);
    pot5Val = analogRead(pot5);
    pot5Angle = map(pot5Val, 0, 1023, 0, 180);
    pot6Val = analogRead(pot6);
    pot6Angle = map(pot6Val, 0, 1023, 0, 180);

    stepper.runToNewPosition(pot1Angle);
    stepper2.runToNewPosition(pot2Angle);
    stepper3.runToNewPosition(pot3Angle);
    servo1.write(pot4Angle);
    servo2.write(pot5Angle);
    servo3.write(pot6Angle);
}
```

Codefragment 7: Definiëren variabelen voor posities [32] [33]

In het volgende deel wordt er gekeken of de knop om op te slaan wordt ingedrukt. Als dat zo is, wordt de buttonState laag en is er verandering te zien. Hierdoor wordt buttonPushCounter +1 gedaan en licht de led heel kort op (Codefragment 8). Dit betekent dat de posities van de potentiometers worden opgeslagen.

```
//if button1 is pressed (HIGH), save the potentiometers position
//as long as button1 is pressed
buttonState1 = digitalRead(buttonPin1);

if (buttonState1 != lastButtonState1) {

    if (buttonState1 == LOW) {
        buttonPushCounter++;
        digitalWrite(led1, HIGH);
        delay(100);
        digitalWrite(led1, LOW);
    }
    lastButtonState1 = buttonState1;
}
```

Codefragment 8: Indrukken van de save knop

In de onderstaande code wordt er gekeken of de startknop wordt ingedrukt (Codefragment 9). Als dat zo is, worden alle posities constant na elkaar uitgevoerd en wordt er repetitief werk gedaan. Er brandt eveneens een led die het repetitieve werk aangeeft. (Codefragment 10).

```
buttonState2 = digitalRead(buttonPin2);

if (buttonState2 != lastButtonState2) {

    if (buttonState2 == LOW) {
        button2Pressed = true;
        delay(50);
    }
    lastButtonState2 = buttonState2;
}
```

Codefragment 9: Indrukken van de save knop

```
if (button2Pressed) {
    digitalWrite(led2, HIGH);
    for (int i = 0; i < 8; i++) {
        stepper.runToNewPosition(stepper1PosSave[i]);
        stepper2.runToNewPosition(stepper2PosSave[i]);
        stepper3.runToNewPosition(stepper3PosSave[i]);
        servo1.write(servo4PosSave[i]);
        servo2.write(servo5PosSave[i]);
        servo2.write(servo6PosSave[i]);
    }
}
```

Codefragment 10: Indrukken van de save knop [32] [33]

Dit is de functie die in het begin van de code werd aangemaakt bij het instellen van de interruptpin. Als stopknop wordt ingedrukt, stopt de loop om alle stappen na elkaar uit te voeren. De counter wordt terug op nul gezet om opnieuw te beginnen en de led gaat uit (Codefragment 11).

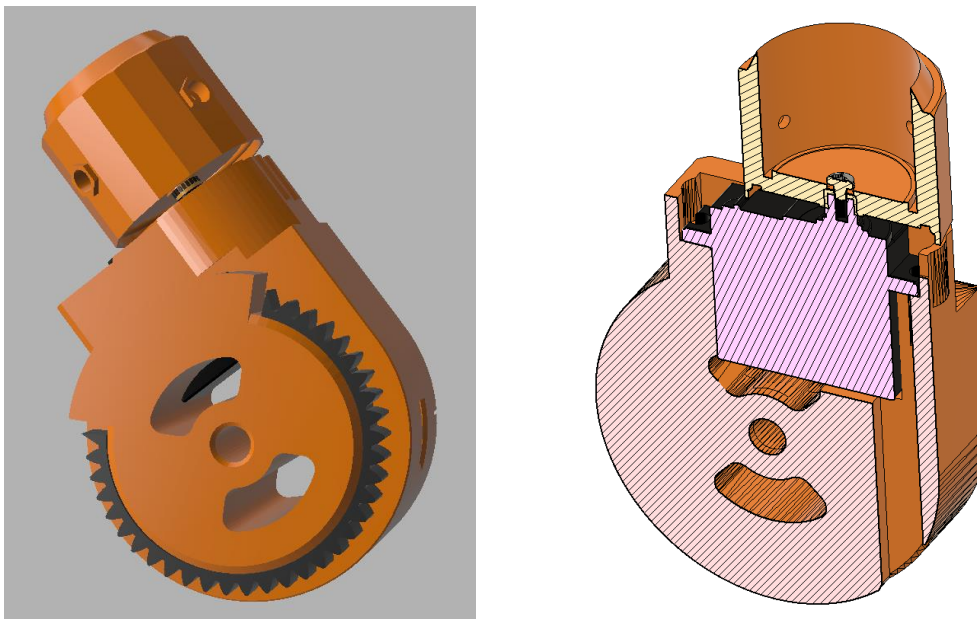
```
void buttonISR() {  
    button2Pressed = false;  
    buttonPushCounter = 0;  
    digitalWrite(led2, LOW);  
}
```

Codefragment 11: Indrukken van de stop knop

3.3 Motoren

Zoals in de onderzoeksfase al werd beschreven, maakt de arm gebruik van twee types motoren: servo- en stappenmotoren, waarvan de voordelen werden besproken. De grote uitdaging in de realisatie is het samenvoegen van de motor en de 3D-prints.

In de ontwerpfase werd uitgelegd dat over de as van de stappenmotor een tandwiel kan worden geschoven. Bij de servomotor is het echter iets moeilijker. Bij dat type worden motoras en onderdeel mechanisch met elkaar verbonden. Beide oppervlakken moeten dus perfect op elkaar zijn afgestemd.



Figuur 35: Volledige vierde as met servomontage en doorsnede met analyse

In de 3D-print wordt een raakoppervlak voorzien waar de tandjes op de as van de servomotor kunnen bijten wanneer de schroef wordt aangedraaid. Op die manier draait het onderdeel mee wanneer de servomotor een beweging maakt. Deze methode wordt verduidelijkt in Figuur 35. Hier is de manier waarop de motor is gemonteerd te zien. De kracht van het hedendaags 3D-ontwerp is zichtbaar. Alle componenten kunnen worden toegevoegd om de kans op onaangename verrassingen te voorkomen.

4 Testen

De pcb is volledig af en is klaar om uitgebreid te worden getest met focus op de hardware. Eerst worden alle connecties gecontroleerd op kortsluitingen, dan worden pinnen uitgemeten om te kijken of de correcte spanning terug te vinden is. Dit moet kloppen met de vooropgestelde verwachtingen die terug te vinden zijn in de datasheet. (Bijlage 1, Bijlage 2, Bijlage 3)

Als volgt worden alle componenten getest en stuk voor stuk uitgelezen in een testprogramma. Dit gebeurt met behulp van de Arduino IDE. Er wordt gebruik gemaakt van verschillende bibliotheken om alles te testen.

4.1 Connecties

Om te testen of alle hardware connecties in orde zijn, wordt gebruik gemaakt van een multimeter. Hiermee wordt de weerstand tussen twee punten gemeten, waardoor kortsluitingen gedetecteerd worden. Een toon is te horen bij het verbinden van twee dezelfde signalen. Verder worden voedingspinnen en datalijnen uitgemeten om te controleren of de juiste spanning over deze pinnen staat. Hiervoor is een DC-voltagestand aanwezig op het toestel.

Na het solderen, viel direct op dat er een kortsluiting mogelijk is tussen de voedingspin en de groundpin onder de CP2102 en TMC2208. Deze kortsluiting is makkelijk gemaakt doordat dit een heel klein component is. Teveel soldeerpaste is hiervan de oorzaak, zo vloeit het soldeer onder het component en is er dus meer kans op kortsluiting.

Om deze fouten recht te zetten, wordt een hetelucht-soldeerstation gebruikt. Deze blaast hete lucht met een temperatuur rond de 300 graden Celsius zodat het soldeersel terug opgewarmd en vloeibaar wordt. Nu is de kans op kortsluiting bij het aansluiten al een pak minder.

4.2 Werking

Nu de kans kleiner is op een kortsluiting, is het moment aangebroken om spanning op de pcb te steken en te testen.



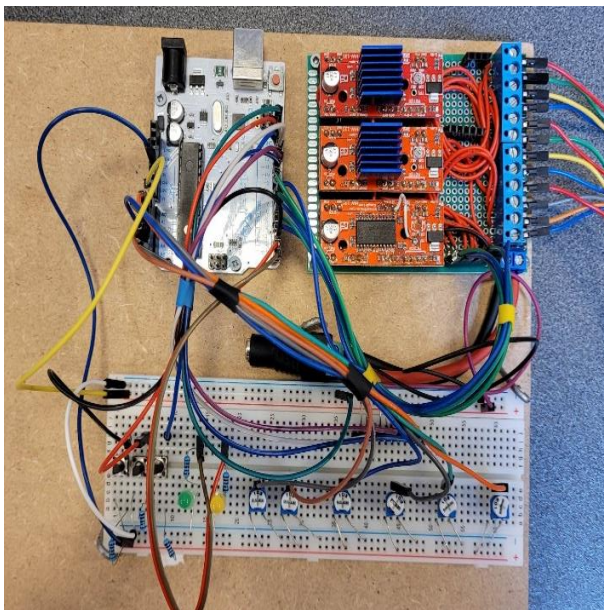
Figuur 36: TMC2208 risico

Bij het uploaden met de USB-C-poort moet het apparaat tevoorschijn komen in het apparaatbeheer van een pc. Dit verscheen niet op de pc. De CP2102 is mogelijk defect door het opnieuw opwarmen met het hetelucht-soldeerstation. Een alternatief was al voorzien op de print, dus was het vanzelfsprekend om voor de optie te gaan met externe CP2102 of FTDI. Figuur 36

Na een tijd zoeken, werd snel duidelijk dat dit kon worden geprogrammeerd. Tijdens het uploaden blijft de bootknop ingedrukt (GPIO0) als er continue op het computerscherm komt, pas dan kan de resetknop worden gebruikt. Als dit gebeurd is, is er een percentage zichtbaar op het scherm. Indien dit 100% is, kan de bootknop worden losgelaten.

Het proces voor het uploaden van het programma kan enkel als de stroom niet aangesloten is. Als dit wel het geval is, is er een error aanwezig: "fatal error occurred: Serial data stream stopped: Possible serial noise or corruption."

De bedoeling was om drie stappenmotoren en drie servomotoren aan te sturen. De eerder besproken td-8125mg trokken te veel stroom, dit is dan ook de reden om MR 996R te gebruiken. De stappenmotoren zijn wel hetzelfde gebleven.



Figuur 37: Back-upplan

Bij het testen op de print, ontplofte de LDO en was de chip ook kapot. Na enkele metingen en testen was het duidelijk dat er een fout was bij het stroomverbruik van de motor. De pinnen op de print leveren maar 5 V i.p.v. 12 V. Dit is iets om zeker rekening met te houden bij een vervolgontwerp.

Doordat de CP2102 niet werkte, is er voor gekozen om de TMC2208 niet aan te sturen. Het was te risicovol omdat er nog maar één reservechip op overschot was. Daarom is er nog een back-upplan voorzien. Figuur 37 Om te testen is er gebruikt gemaakt van de datasheets Bijlage 1, Bijlage 2, Bijlage 3.

4.3 Veiligheid

Veiligheid is in dit type projecten een erg belangrijke factor. Nog meer wanneer een prototype wordt omgezet in een verkoopbaar product. Om dit prototype zo realistisch mogelijk te maken, werden een aantal veiligheidsaspecten opgenomen in het ontwerp.

Zoals in de foto's van de ontwerpen is te zien, werd de arm vervaardigd uit een oranje gekleurd materiaal. In een kleine testomgeving maakt het kleur weinig verschil. Wanneer de arm wordt meegenomen naar een industriële omgeving is dit aspect meteen veel belangrijker. Hier kunnen meerdere werknemers zich rondom de arm bewegen. De oranje kleur is in deze omgevingen erg goed zichtbaar en op die manier kunnen bewegingen door werknemers goed worden ingeschat en ongelukken worden voorkomen.

De arm werd ook elektronisch beveiligd. Eén van de voorbeelden daarvan is de stroomlimiet die op de voeding zit. Op die manier kan de arm nooit onverwachte bruuske bewegingen maken. Probeert hij dat toch te doen, dan krijgt hij simpelweg de stroom niet om de beweging te starten.

Een tweede voorbeeld zijn de veiligheidsvoorschriften op de pcb. Zoals al meermaals werd besproken, wordt gebruik gemaakt van stappenmotoren die gestuurd worden door een driver. Deze drivers kunnen door het snel sturen erg warm worden, daarom werd ook op de print het correcte symbool aangebracht dat waarschuwt voor een warm oppervlak. Figuur 38



Figuur 38: Veiligheidsicoon hitte

4.4 Functionaliteit

Dit deel richt zich op de bespreking van de functionaliteit van de robotarm. Functionaliteit in deze context wordt gedefinieerd als niet alleen het basismechanisme en de werking van de robotarm, maar ook de mate waarin het systeem erin slaagt om te voldoen aan de verwachte prestatienormen en biedt een hoog niveau van gebruiksgemak voor de gebruikers.

Het ontwerp van de robotarm omvat verschillende unieke eigenschappen die bijdragen aan de functionaliteit ervan. De motoren die in het ontwerp worden gebruikt, spelen een cruciale rol in het leveren van de benodigde mechanische output. Ze hebben de capaciteit om de robotarm te verplaatsen in de gewenste richting, een functie die van vitaal belang is voor de meeste, zo niet alle, toepassingen van de robotarm. De precisie van deze bewegingen, evenals de snelheid waarmee ze worden uitgevoerd, worden gereguleerd door het gekozen type motor en het besturingssysteem dat in het ontwerp is geïntegreerd.

Het opzetstuk is een ander onderdeel dat bijdraagt aan de functionaliteit van de robotarm. De keuze van het opzetstuk hangt grotendeels af van de specifieke toepassing van de robotarm en kan dus variëren. Afhankelijk van het specifieke opzetstuk, kan de robotarm een breed scala aan taken uitvoeren. Dit kan variëren van eenvoudige pick-and-place operaties wat handig is in bijvoorbeeld assemblage of sorteerlijnen tot meer complexe taken zoals lassen, boren of zelfs 3D-printen.

Tenslotte speelt de programmeerbaarheid van het apparaat een cruciale rol in de algehele functionaliteit. De mogelijkheid om de bewegingen en gedragingen van de robotarm te programmeren, maakt het mogelijk om het apparaat aan te passen aan een breed scala van taken en omgevingen. Het biedt een aanzienlijk niveau van flexibiliteit in de toepassing van de robotarm, waardoor de robotarm veelzijdiger en bruikbaar wordt in verschillende industrieën en voor verschillende doeleinden. Deze programmeerbaarheid stelt gebruikers ook in staat om het systeem te optimaliseren op basis van hun specifieke behoeften, waardoor het potentieel voor efficiëntie en productiviteit wordt verhoogd.

4.5 Programmatie

Om te beginnen met het testen van de code werd dat stap voor stap gedaan. Eerst het sturen van de stappenmotoren en servomotoren via de potentiometers. Dan langzaam aan meer code toevoegen en testen of het werkt.

Een voorbeeld van testen is de interruptie (Codefragment 12). Om midden in de code te stoppen is er gebruikt gemaakt van interruptie. Dat betekent waar dan ook dat de cyclus in de code zit, het zal eerst de code van de interruptieknop uitvoeren. Hieronder is een voorbeeld dat een led aan en uit zet als er op de knop wordt gedrukt. Deze is getest op een Arduino Uno. Enkel digitale pinnen 2, 3 hebben een interruptie bij de Arduino Uno. Hiermee werd rekening gehouden.

```
// Pin connected to the interrupt
const int interruptPin = 2;

// Interrupt service routine
void handleInterrupt() {
    digitalWrite(LED_BUILTIN, HIGH); // Turn on the LED
    delay(500); // Keep the LED on for 500 milliseconds
    digitalWrite(LED_BUILTIN, LOW); // Turn off the LED
}

void setup() {
    // Attach interrupt to the pin
    attachInterrupt(digitalPinToInterrupt(interruptPin), handleInterrupt,
CHANGE);

    // Set the LED pin as an output
    pinMode(LED_BUILTIN, OUTPUT);
}

void loop() {
    // Delay for a short period
    delay(500);
}
```

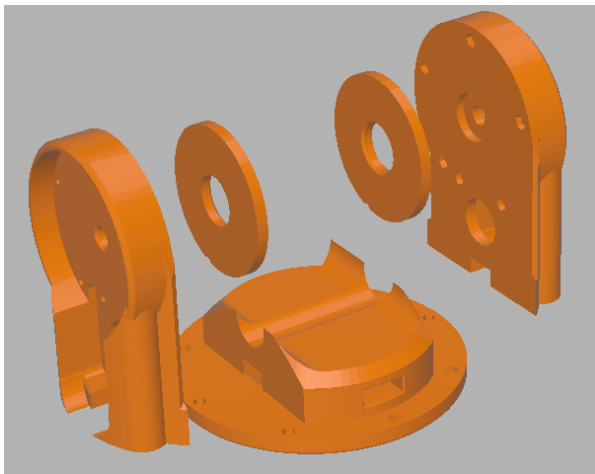
Codefragment 12: Indrukken van de save knop

4.6 Mechanische onderdelen

In het mechanisch ontwerp vallen realisatie en testen steeds samen. Dit omdat met 3D-printen snel een aanpassing aan een onderdeel kan worden gemaakt. Aandachtspunten zoals toleranties en het correct passen van onderdelen kunnen via meerdere versies worden geperfectioneerd. Ook hier is de kracht van 3D-ontwerp weer erg opvallend. Toleranties kunnen worden toegevoegd en bij een nieuwe versie makkelijk worden aangepast. Dat zonder het volledige design opnieuw te tekenen. Motoren kunnen rechtstreeks worden toegevoegd om de kans op fouten te verkleinen.

De tandwielen pasten niet meteen op de assen van de stappenmotoren. De tolerantiewaarden werden aangepast en enkele tijd later kon het prototype worden opgebouwd.

Wanneer meerdere versies van grote onderdelen worden geprint, is denken aan de manier waarop ze worden geprint een erg belangrijk detail. Dit is ecologisch in twee richtingen: er is minder ondersteunend materiaal nodig en het hele onderdeel moet niet steeds opnieuw worden geprint. Een mooi voorbeeld hiervan is zichtbaar in Figuur 40. Dit onderdeel werd in meerdere delen opgesplitst om de print met minder ondersteuning en korter uit te voeren. Met als eindresultaat Figuur 39.



Figuur 40: Opsplitsen van een ontwerp



Figuur 39: Eindresultaat robotarm

Conclusie

In dit project is gewerkt aan de opbouw van een robotarm, waarbij gebruik is gemaakt van drie stappenmotoren en drie servomotoren. Voor de stappenmotoren was het noodzakelijk om een driver te gebruiken, waardoor een precieze en gecontroleerde beweging van de arm mogelijk werd gemaakt.

Helaas deden zich problemen voor bij het verbinden van deze motoren, waardoor er een back-upplan moest worden opgesteld. Desondanks is er een eigen 3D-print van de robotarm gemaakt en werden er aparte codes geschreven voor het programma op een ESP32-S2 en het back-upplan op een Arduino Uno.

Het cruciale 3D-ontwerp van de robotarm bood de mogelijkheid om de arm nauwkeurig en efficiënt te construeren, met de vrijheid om aanpassingen te maken die aansluiten bij het specifieke doel. Het hoofddoel van het project was om de robotarm zelfstandig een cyclus te laten doorlopen nadat de posities waren bepaald. Ondanks enkele uitdagingen is er uiteindelijk een werkend prototype gecreëerd.

De chiptekorten sinds 2022, vooral na de coronaperiode, hebben de kosten van het project beïnvloed. Ondanks deze stijging van de prijzen is het gelukt om dit relatief goedkoop te houden door te bestellen bij AliExpress en is het project succesvol afgerond.

Er wordt waardering uitgesproken naar de Technologiecampus Odisee voor de waardevolle ondersteuning tijdens dit project, evenals naar medestudenten en ouders die hun bijgedragen en steun hebben geboden om dit project mogelijk te maken. Het project heeft waardevolle ervaring en kennis opgeleverd op het gebied van robotica, 3D-printen en programmeren, waarmee verdere verbeteringen aan de robotarm kunnen worden gerealiseerd en nieuwe uitdagingen in de toekomst kunnen worden aangegaan. Enkele mogelijke verbeteringen zouden kunnen bestaan uit het aanpassen van de grootte van de chips, zodat ze gemakkelijk kunnen worden gesoldeerd.

In dit project is de waardevolle ondersteuning van ChatGPT, een geavanceerd AI-model ontwikkeld door OpenAI, geraadpleegd via de volgende link: <https://chat.openai.com/>. De toegankelijkheid van dit AI-model en de waardevolle bijdrage die het heeft geleverd aan het project worden zeer gewaardeerd.

Over het geheel genomen is er trots op wat er is bereikt en wordt er uitgekeken naar nieuwe projecten en mogelijkheden om vaardigheden verder te ontwikkelen.

Literatuurlijst

- [1] S. P. Dinkgreve, „wetenschapsschool,” Natuurkunde voor de Middelbare School, 2018. [Online]. Available: <https://ap.lc/MHHQ6>. [Geopend 2023 maart 29].
- [2] „Domoticx,” [Online]. Available: <https://ap.lc/deZqp>. [Geopend 17 Maart 2023].
- [3] „Tinytronics,” [Online]. Available: <https://ap.lc/eTqNZ>. [Geopend 17 maart 2023].
- [4] „Tinytronics,” [Online]. Available: <https://ap.lc/3loJv>. [Geopend 17 maart 2023].
- [5] „Conrad,” [Online]. Available: <https://ap.lc/pZaYL>. [Geopend 17 maart 2023].
- [6] „Arduino,” [Online]. Available: <https://ap.lc/OHA2n>. [Geopend 18 maart 2023].
- [7] „Tinytronics,” [Online]. Available: <https://ap.lc/gQWu6>. [Geopend 18 maart 2023].
- [8] „Digikey,” [Online]. Available: <https://ap.lc/i7Pd2>. [Geopend 20 maart 2023].
- [9] C. B.-S. 4. Chindi.ap, „Arrow,” 2018 november 19. [Online]. Available: <https://ap.lc/Hhr6j>. [Geopend 2023 maart 28].
- [10] „Mouser,” STMicroelectronics, 2021. [Online]. Available: <https://ap.lc/08Oq4>. [Geopend 20 maart 2023].
- [11] „Mchobby,” [Online]. Available: <https://ap.lc/jClrO>. [Geopend 17 maart 2023].
- [12] M. K. C. G. M. Dhruvi Ranjan Gaan, „semanticscholar,” 2017. [Online]. Available: <https://ap.lc/3FZnb>. [Geopend 2023 maart 28].
- [13] „Alldatasheet,” [Online]. Available: <https://ap.lc/CW7Hs>. [Geopend 17 maart 2023].
- [14] „Robotic arms,” [Online]. Available: <https://ap.lc/cL01R>. [Geopend 21 maart 2023].
- [15] R. Vaughn, „Machinedesign,” Machinedesign, 2 december 2013. [Online]. Available: <https://ap.lc/Xp3Y1>. [Geopend 21 maart 2023].

- [16] „Go to shopping,” [Online]. Available: <https://ap.lc/ZoN4D>. [Geopend 21 maart 2023].
- [17] M. Claussen, „Elektormagazine,” Elektormagazine, 12 juli 2022. [Online]. Available: <https://ap.lc/lHw1e>. [Geopend 21 maart 2023].
- [18] „Kuka,” Kuka, [Online]. Available: <https://ap.lc/gyjYQ>. [Geopend 21 maart 2023].
- [19] T. Nardi, „Hackaday,” Hackaday, 19 oktober 2020. [Online]. Available: <https://ap.lc/UFSXb>. [Geopend 21 maart 2023].
- [20] G. Klop, „Pure.tue,” Tue, 1992 januari 01. [Online]. Available: <https://ap.lc/0subo>. [Geopend 2023 maart 29].
- [21] Sparkfun, „Sparkfun,” Sparkfun, [Online]. Available: <https://ap.lc/GlkWh>. [Geopend 19 mei 2023].
- [22] Espressif, „Espressif,” Espressif systems, 2023. [Online]. Available: <https://ap.lc/W5mUX>. [Geopend 19 mei 2023].
- [23] TRINAMIC, „Digikey,” Digikey, 12 2017. [Online]. Available: <https://www.digikey.be/reference-designs/en/motor-control/motor-control/2276>. [Geopend 18 mei 2023].
- [24] Espressif, „Espressif,” Espressif systems, 06 December 2012. [Online]. Available: <https://ap.lc/yFm0c>. [Geopend 20 mei 2023].
- [25] Blaisebaette, „Instructables,” [Online]. Available: <https://ap.lc/KWCGc>. [Geopend 20 mei 2023].
- [26] G. Schorcht, „Riot,” [Online]. Available: <https://ap.lc/q2g6g>. [Geopend 20 mei 2023].
- [27] Robotics, „robotics,” [Online]. Available: <https://ap.lc/yF6s1>. [Geopend 21 mei 2023].
- [28] Pishop, „Pishop,” [Online]. Available: <https://ap.lc/FYUwR>. [Geopend 22 mei 2023].
- [29] Kauda, „Kauda,” [Online]. Available: <https://www.diy-tech.it/010-kauda>. [Geopend 1 05 2023].

- [30] TME, „TME,” TME, 08 augustus 2020. [Online]. Available: <https://ap.lc/VQVPt>. [Geopend 22 mei 2023].
- [31] „circuitcalculator,” 31 januari 2006. [Online]. Available: <https://ap.lc/aBSjM>. [Geopend 23 mei 2023].
- [32] Brainy-Bits, „Brainy-Bits,” 23 10 2020. [Online]. Available: <https://ap.lc/NNVai>. [Geopend 18 05 2023].
- [33] M. McCauley, „airspayce,” 2010. [Online]. Available: <https://ap.lc/lcZLI>. [Geopend 18 05 2023].
- [34] G. Brandl, „pygments,” 2006. [Online]. Available: <https://ap.lc/M75In>. [Geopend 28 05 2023].
- [35] Y. YAKIMOVA, „Europees parlement,” Europees parlement, 07 juni 2022. [Online]. Available: <https://ap.lc/BxmRq>. [Geopend 20 maart 2023].
- [36] „almotion,” almotion, [Online]. Available: <https://ap.lc/amwzP>. [Geopend 2023 maart 28].
- [37] „zvstechniek,” zvstechniek, [Online]. Available: <https://ap.lc/gdEbe>. [Geopend 2023 maart 28].

Bijlagenoverzicht

Bijlage 1: Kopieën datasheets

ESP32-S2 Series Datasheet

SoC with Xtensa® Single-Core 32-bit LX7 Microprocessor
Supporting IEEE 802.11b/g/n (2.4 GHz Wi-Fi)

Including:

ESP32-S2
ESP32-S2FH2
ESP32-S2FH4
ESP32-S2FN4R2
ESP32-S2R2



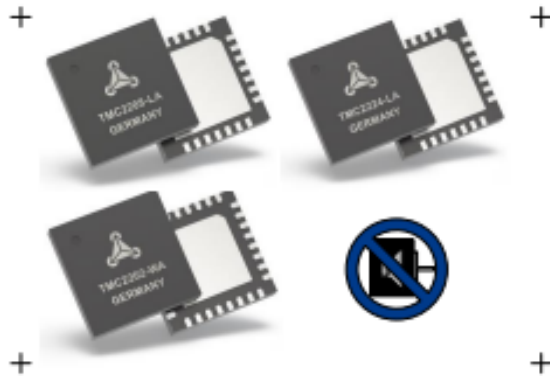
Version 1.5
Espressif Systems
Copyright © 2023



www.espressif.com

TMC2208/2 & TMC2224 family Datasheet

TMC2202, TMC2208, TMC2224 Step/Dir Drivers for Two-Phase Bipolar Stepper Motors up to 2A peak - StealthChop™ for Quiet Movement - UART Interface Option.



APPLICATIONS

Compatible Design Upgrade
3D Printers
Printers, POS
Office and home automation
Textile, Sewing Machines
CCTV, Security
ATM, Cash recycler
HVAC

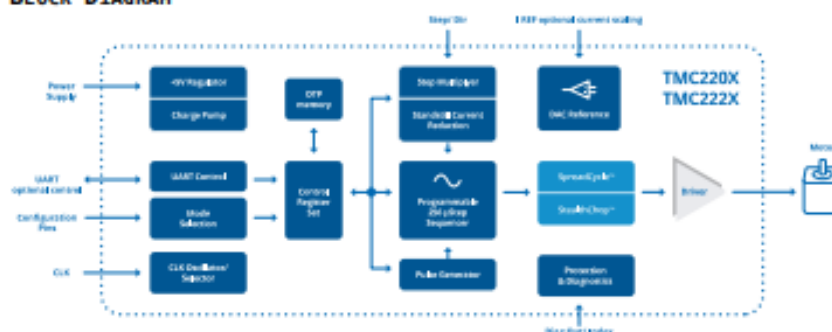
FEATURES AND BENEFITS

2-phase stepper motors up to 2A coil current (peak)
STEP/DIR Interface with 2, 4, 8, 16 or 32 microstep pin setting
Smooth Running 256 microsteps by **MicroPlyer™** interpolation
StealthChop2™ silent motor operation
SpreadCycle™ highly dynamic motor control chopper
Low RDSon LS 280mΩ & HS 290mΩ (typ. at 25°C)
Voltage Range 4.75... 36V DC
Automatic Standby current reduction (option)
Internal Sense Resistor option (no sense resistors required)
Passive Braking and Freewheeling
Single Wire UART & OTP for advanced configuration options
Integrated Pulse Generator for standalone motion
Full Protection & Diagnostics
Choice of QFN and wettable QFN packages for best fit

DESCRIPTION

The TMC2202, TMC2208 and TMC2224 are ultra-silent motor driver ICs for two-phase stepper motors. Their pinning is compatible to a number of legacy drivers. TRINAMICs sophisticated StealthChop2 chopper ensures noiseless operation, maximum efficiency and best motor torque. Its fast current regulation and optional combination with SpreadCycle allow for highly dynamic motion. Integrated power-MOSFETs handle motor current up to 1.4A RMS. Protection and diagnostic features support robust and reliable operation. A simple to use UART interface opens up more tuning and control options. Application specific tuning can be stored to OTP memory. Industries' most advanced STEP/DIR stepper motor driver family upgrades designs to noiseless and most precise operation for cost-effective and highly competitive solutions.

BLOCK DIAGRAM



TRINAMIC Motion Control GmbH & Co. KG
Hamburg, Germany



SINGLE-CHIP USB-TO-UART BRIDGE

For newer designs, the CP2102N devices offer compatible footprints and are recommended for use instead of the CP2102/9. See the Silicon Labs website (www.silabs.com/usbxpress) for more information.

Single-Chip USB to UART Data Transfer

- Integrated USB transceiver; no external resistors required
- Integrated clock; no external crystal required
- Internal 1024-byte programmable ROM for vendor ID, product ID, serial number, power descriptor, release number, and product description strings
 - EEPROM (CP2102)
 - EPROM (One-time programmable) (CP2109)
- On-chip power-on reset circuit
- On-chip voltage regulator
 - 3.3 V output (CP2102)
 - 3.45 V output (CP2109)
- 100% pin and software compatible with CP2101

USB Function Controller

- USB Specification 2.0 compliant; full-speed (12 Mbps)
- USB suspend states supported via SUSPEND pins

Asynchronous Serial Data BUS (UART)

- All handshaking and modem interface signals
- Data formats supported:
 - Data bits: 5, 6, 7, and 8
 - Stop bits: 1, 1.5, and 2
 - Parity: odd, even, mark, space, no parity
- Baud rates: 300 bps to 1 Mbps
- 576 Byte receive buffer; 640 byte transmit buffer
- Hardware or X-On/X-Off handshaking supported
- Event character support
- Line break transmission

Virtual COM Port Device Drivers

- Works with existing COM port PC Applications
- Royalty-free distribution license
- Windows 8/7/Vista/Server 2003/XP/2000
- Mac OS-X/OS-9
- Linux

USBXpress™ Direct Driver Support

- Royalty-Free Distribution License
- Windows 7/Vista/XP/Server 2003/2000
- Windows CE

Example Applications

- Upgrade of RS-232 legacy devices to USB
- Cellular phone USB interface cable
- USB interface cable
- USB to RS-232 serial adapter

Supply Voltage

- Self-powered: 3.0 to 3.6 V
- USB bus powered: 4.0 to 5.25 V

Package

- RoHS-compliant 28-pin QFN (5x5 mm)

Ordering Part Numbers

- CP2102-GM
- CP2109-A01-GM

Temperature Range: -40 to +85 °C

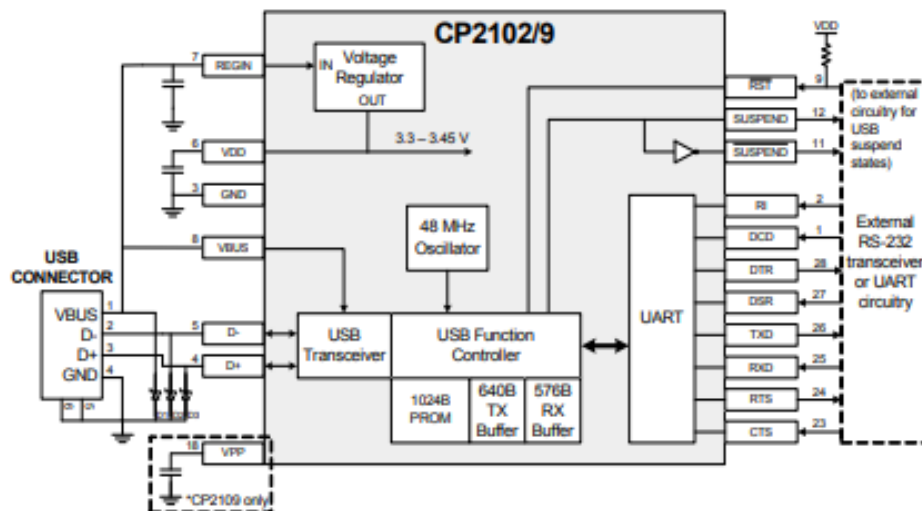


Figure 1. Example System Diagram

Bijlage 2: BOM

Tabel 5: BOM

Product	Type	Link	Stuks	Prijs	Prijs/stuk	Gebruikte stuks	Totaalprijs per gebruikte stuks
Voedingsstekker	SMD	https://nl.aliexpress.com/item/4000677508616.html?spm=a2g0o.order_list.order_list_main.35.7b2f79d2tsYecZ&gatewayAdapt=glo2nld	10	€ 1,44	€ 0,144	1	€ 0,144
CP2102	SMD	https://nl.aliexpress.com/item/33000297490.html?spm=a2g0o.order_list.order_list_main.41.7b2f79d2tsYecZ&gatewayAdapt=glo2nld	3	€ 4,47	€ 1,49	1	€ 1,49
USB-C-poort	SMD	https://nl.aliexpress.com/item/1005004804811591.html?spm=a2g0o.order_list.order_list_main.47.7b2f79d2tsYecZ&gatewayAdapt=glo2nld	10	€ 0,54	€ 0,054	1	€ 0,054
Jumper	THT	https://nl.aliexpress.com/item/1005004985941167.html?spm=a2g0o.order_list.order_list_main.59.7b2f79d2tsYecZ&gatewayAdapt=glo2nld	100	€ 0,74	€ 0,0074	2	€ 0,0148
Rotary encoder	THT	https://nl.aliexpress.com/item/10000056483250.html?spm=a2g0o.order_list.order_list_main.64.7b2f79d2tsYecZ&gatewayAdapt=glo2nld	5	€ 0,71	€ 0,142	1	€ 0,142
Pottrimmer voor TMC2208	SMD	https://nl.aliexpress.com/item/1005002763769561.html?spm=a2g0o.order_list.order_list_main.82.7b2f79d2tsYecZ&gatewayAdapt=glo2nld	20	€ 1,81	€ 0,0905	3	€ 0,2715
Meetweerstand voor TMC2208	SMD	https://nl.aliexpress.com/item/1005003411558149.html?spm=a2g0o.order_list.order_list_main.83.7b2f79d2tsYecZ&gatewayAdapt=glo2nld	100	€ 0,54	€ 0,0054	3	€ 0,0162

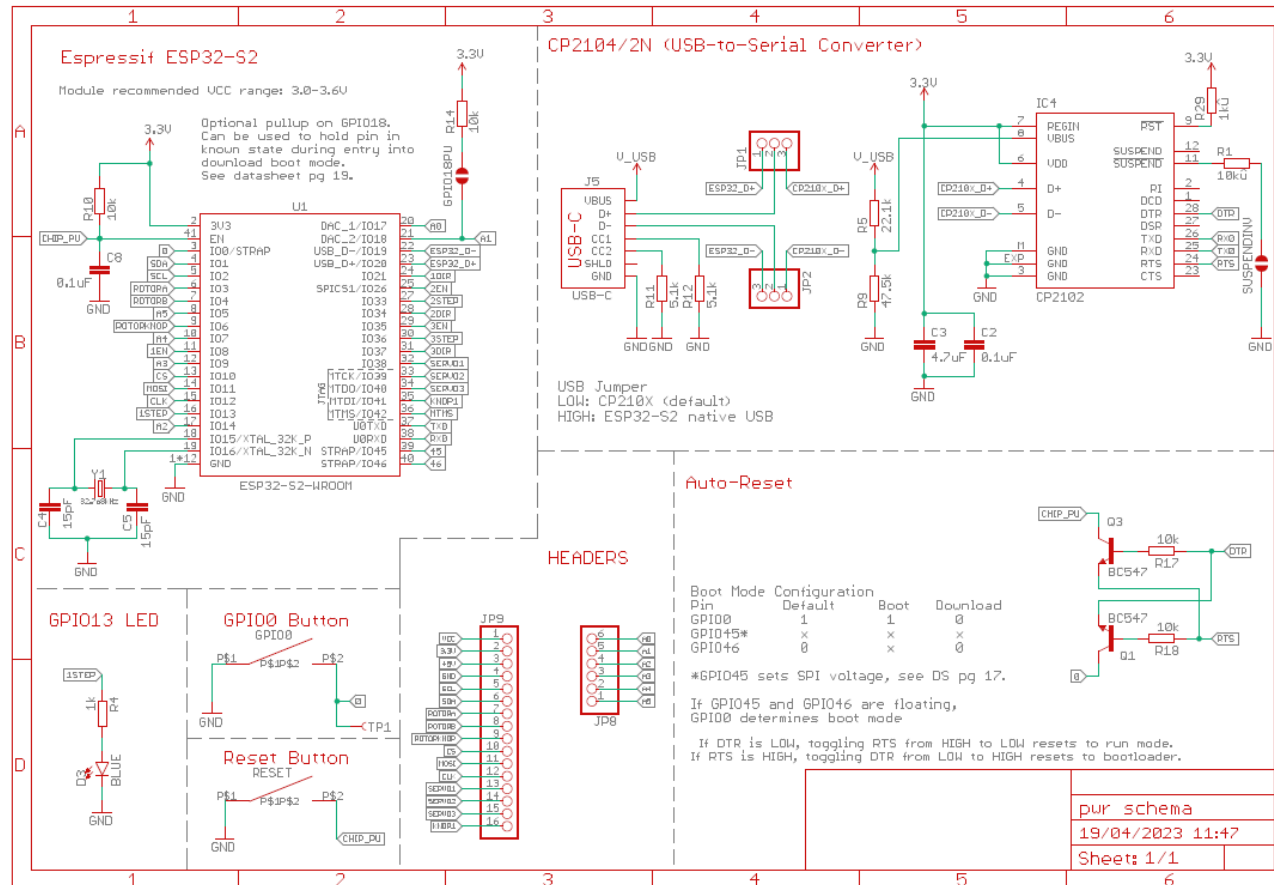
Leds	SMD	https://nl.aliexpress.com/item/1005003327360045.html?spm=a2g0o.order_list.order_list_main.101.7b2f79d2tsYecZ&gatewayAdapt=glo2nld	100	€ 2,28	€ 0,0228	4	€ 0,08912
5 V-LDO	SMD	https://nl.aliexpress.com/item/1005004908874511.html?spm=a2g0o.order_list.order_list_main.122.7b2f79d2tsYecZ&gatewayAdapt=glo2nld	50	€ 1,47	€ 0,0294	1	€ 0,0294
3.3 V-LDO	SMD	https://nl.aliexpress.com/item/1005004908874511.html?spm=a2g0o.order_list.order_list_main.122.7b2f79d2tsYecZ&gatewayAdapt=glo2nld	50	€ 1,47	€ 0,0294	2	€ 0,0588
TMC2208-chip	SMD	https://nl.aliexpress.com/item/1005002873631753.html?spm=a2g0o.order_list.order_list_main.129.7b2f79d2tsYecZ&gatewayAdapt=glo2nld	10	€ 10,24	€ 1,024	3	€ 3,072
ESP32-S2-wroom	SMD	https://nl.aliexpress.com/item/1005002221813688.html?spm=a2g0o.order_list.order_list_main.141.7b2f79d2tsYecZ&gatewayAdapt=glo2nld	3	€ 7,23	€ 2,41	1	€ 2,41
Knop	SMD	https://nl.aliexpress.com/item/4000700060759.html?spm=a2g0o.order_list.order_list_main.147.7b2f79d2tsYecZ&gatewayAdapt=glo2nld	10	€ 1,37	€ 0,137	2	€ 0,274
Condensatoren	SMD	https://nl.aliexpress.com/item/4001345918046.html?spm=a2g0o.order_list.order_list_main.51.209079d2DWapJU&gatewayAdapt=glo2nld	720	€ 6,89	€ 0,009	37	€ 0,333
Weerstanden	SMD	https://nl.aliexpress.com/item/1005001461098423.html?spm=a2g0o.order_list.order_list_main.26.209079d2DWapJU&gatewayAdapt=glo2nld	2000	€ 5,02	€ 0,025	33	€ 0,825

lcd ST7920	THT	https://nl.aliexpress.com/item/1005003916378097.html?spm=a2g0o.productlist.main.21.f7f51444aau90J&algo_pvid=357aed06-1a8f-4e04-962d-232323ec6a1f&algo_exp_id=357aed06-1a8f-4e04-962d-232323ec6a1f-10&pdp_npi=3%40dis%21EUR%214.96%214.91%21%21%21%21%21%40211bf14716824140317308166d077e%2112000027459351006%21sea%21BE%21841531276&curPageLogUiid=X9ZGha1inysk	1	€ 4,91	€ 4,91	1	€ 4,91
Servo MG995	THT	https://www.bol.com/be/nl/p/azdelivery-3-x-mg995-micro-digital-servo-motor-voor-rc-robots-helikoptervliegtuigen-compatibel-met-arduino-inclusief-e-book/9300000119725910/?Referrer=ADVNLGOO002013-G-135735706222-S-1679639870575-9300000119725910&gclid=CjwKCAjw9J2iBhBPEiwAErwpebNH9WkOYeBrWzY8cHZ-MVn6p75LLGyoObHuUK76pHnsliEYkcddphoCmzYQAvD_BwE	3	€ 19,99	€ 6,66	3	€ 19,99
BC547	THT	https://nl.aliexpress.com/item/1005003458995298.html?spm=a2g0o.productlist.main.25.194d611bO6wcNu&algo_pvid=4188a2d8-a71d-45e2-837b-feeebd6bb0ec&algo_exp_id=4188a2d8-a71d-45e2-837b-feeebd6bb0ec-12&pdp_npi=3%40dis%21EUR%210.69%210.65%21%21%21%21%21%402122443916824144206786581d06e0%2112000025882263685%21sea%21BE%21841531276&curPageLogUiid=MeJttTnuwZX3	100	€ 0,65	€ 0,0065	2	€ 0,013

Elco	THT	https://nl.aliexpress.com/item/32803428396.html?spm=a2g0o.order_list.order_list_main.60.303079d22by5pV&gatewayAdapt=glo2nld	200	€ 3,81	€ 0,019	2	€ 0,038
		https://nl.aliexpress.com/item/1005003938244847.html?spm=a2g0o.productlist.main.73.2aff49d9TQChN0&algo_pvid=5740f107-9fca-4e52-bd27-a80c1c05cb40&algo_exp_id=5740f107-9fca-4e52-bd27-a80c1c05cb40-36&pdp_npi=3%40dis%21EUR%213.04%211.52%21%21%21%21%21%40211bda9b16824144711724249d076c%2112000027513761698%21sea%21BE%21841531276&curPageLogUi					
Tactile shakelaar	THT	https://nl.aliexpress.com/item/32665922113.html?srcSns=sns_Copy&spreadType=socialShare&bizType=ProductDetail&social_params=20821561299&aff_fcid=b567e40c98d9443286b9f26b04eb34ad-1682416080730-01145-mOC7ie2&tt=MG&aff_fsk=mOC7ie2&aff_platform=default&sk=mOC7ie2&aff_trace_key=b567e40c98d9443286b9f26b04eb34ad-1682416080730-01145-mOC7ie2&shareId=20821561299&businessType=ProductDetail&platform=AE&terminal_id=dce7fe7314bb41a089845f15ca24b83f&afS	20	€ 1,52	€ 0,076	1	€ 0,076
Stappenmotoren 17HS4401	THT	https://nl.aliexpress.com/item/32665922113.html?srcSns=sns_Copy&spreadType=socialShare&bizType=ProductDetail&social_params=20821561299&aff_fcid=b567e40c98d9443286b9f26b04eb34ad-1682416080730-01145-mOC7ie2&tt=MG&aff_fsk=mOC7ie2&aff_platform=default&sk=mOC7ie2&aff_trace_key=b567e40c98d9443286b9f26b04eb34ad-1682416080730-01145-mOC7ie2&shareId=20821561299&businessType=ProductDetail&platform=AE&terminal_id=dce7fe7314bb41a089845f15ca24b83f&afS	3	€ 27	€ 9	3	€ 27
Pcb	BOAR D	https://jlcpcb.com/	5	€ 28,04	€ 5,608	1	€ 5,608
Totaal							€ 66,85882

Bijlage 4: BOM

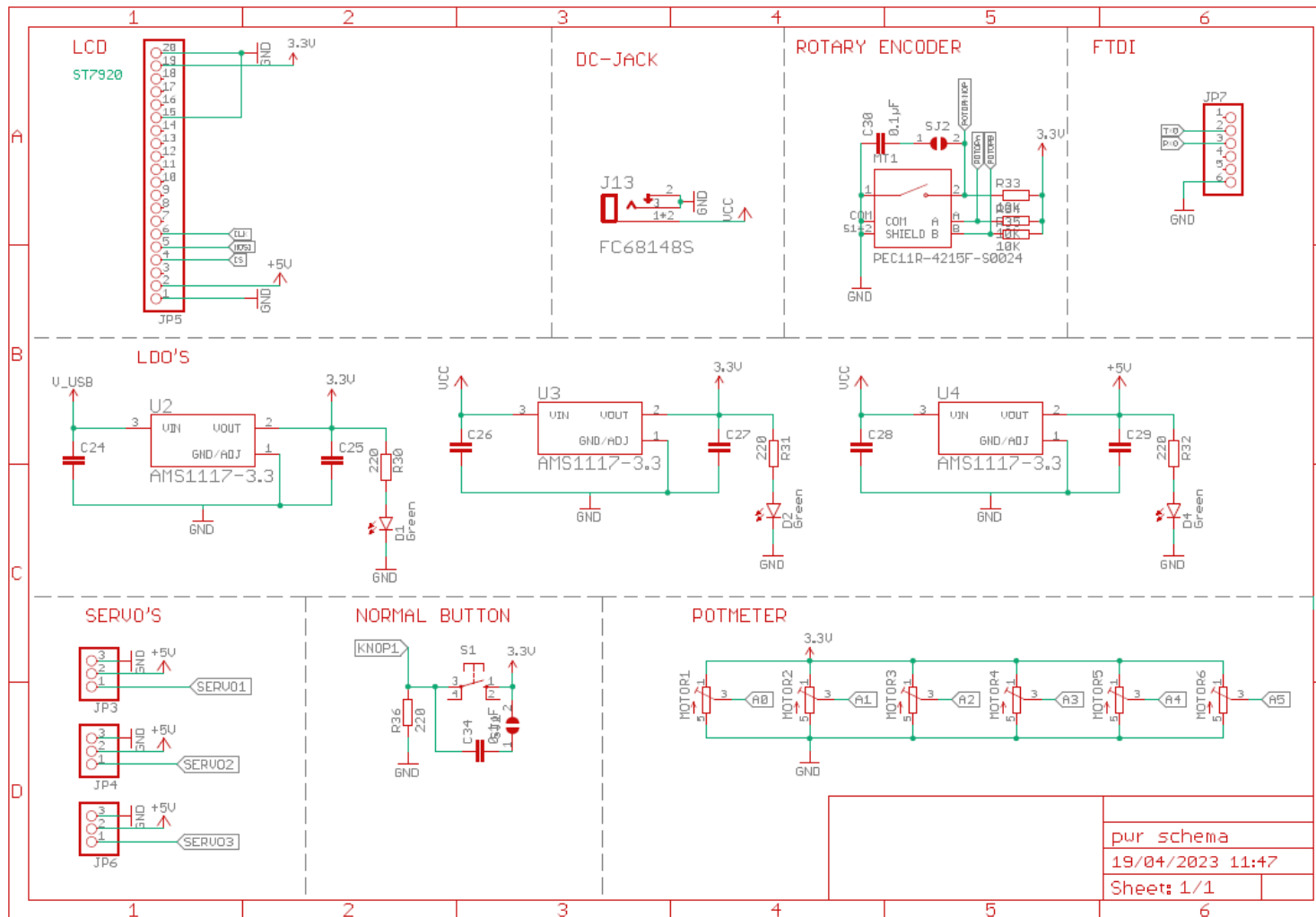
Bijlage 3: Schema



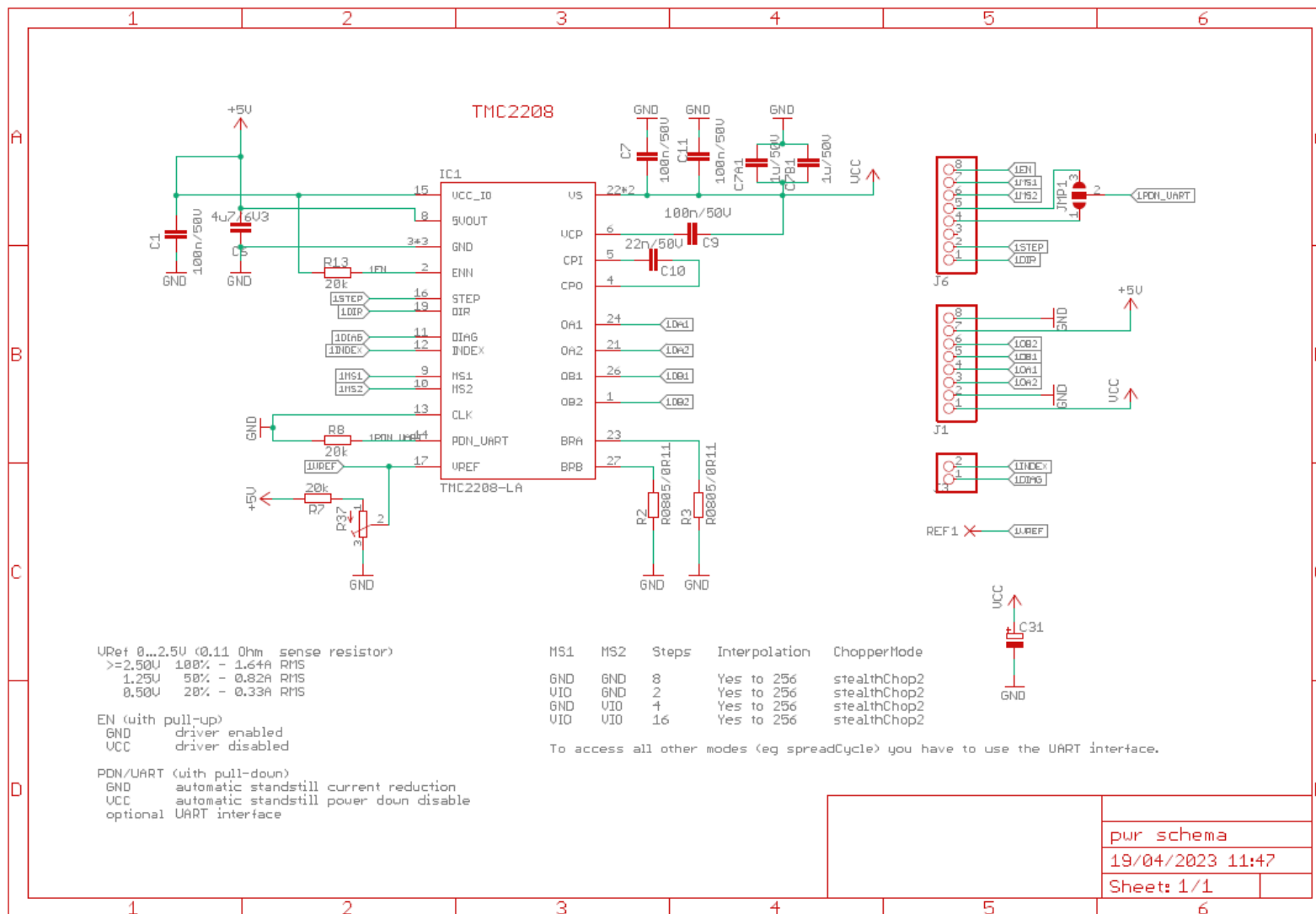
22/04/2023 11:48 C:\Users\Lukas Verschraegen\Downloads\Printplaat voorlopig\V7\pwr schema.sch (Sheet: 1/1)

Bijlage 5: Schema's

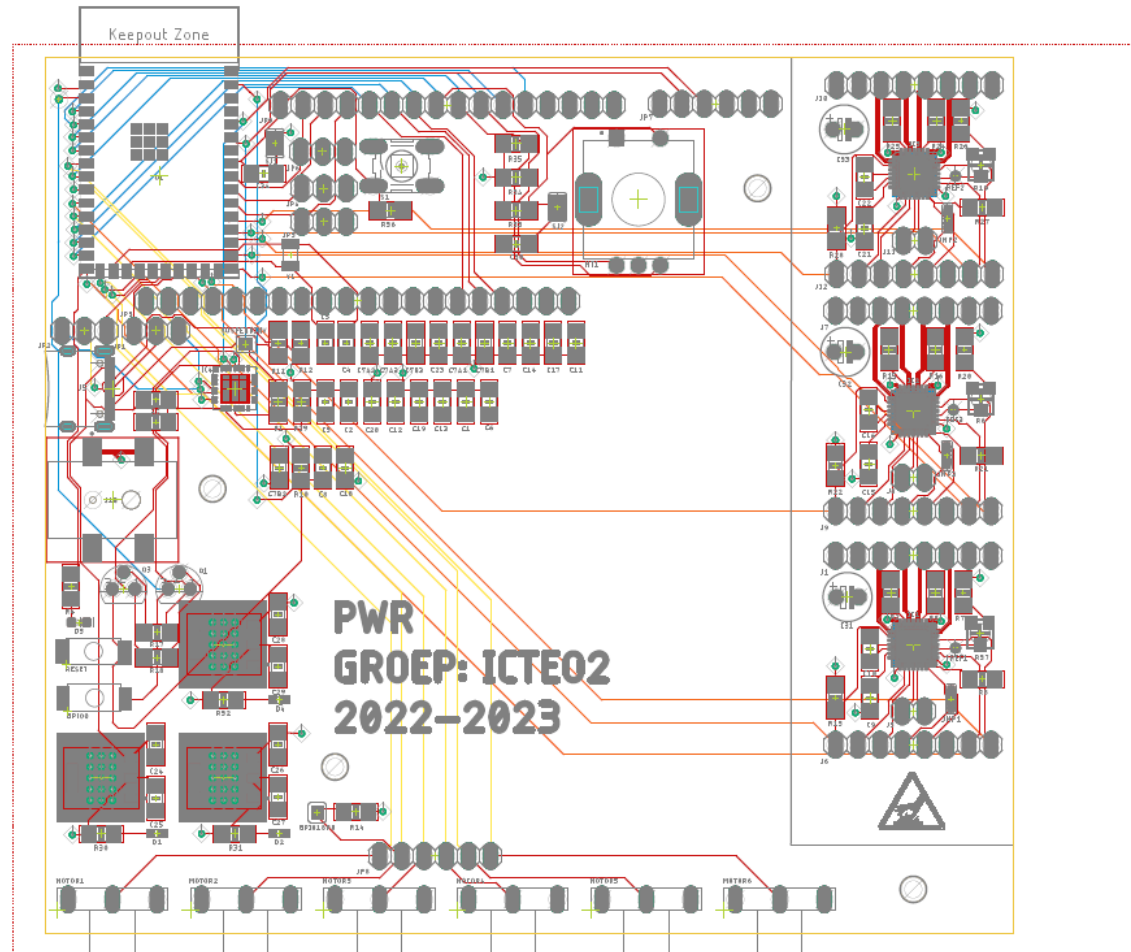
Figuur 41: Schema deel1



Figuur 42: Schema deel2 Bijlage 5



Bijlage 4: Pcb



Bijlage 6: Pcb

Figuur 46: Pcb

Bijlage 5: Logboek rapporteren

Lesweek 7: 13/02 – 19/02

Naam student	Paginnummers in rapport	Taak (bijv. schrijven klad, verbeteren, volledig herschrijven, nakijken, ...)
Lennert De Geeter	Vragenronde op school	Extra informatie vragen om een goed verslag op te stellen.
Roy De Volder	Vragenronde op school	Extra informatie vragen om een goed verslag op te stellen.
Jorik Van Dorpe	Vragenronde op school	Extra informatie vragen om een goed verslag op te stellen.
Lukas Verschraegen	Vragenronde op school	Extra informatie vragen om een goed verslag op te stellen.

Lesweek 8: 20/02 – 26/02

Naam student	Paginnummers in rapport	Taak (bijv. schrijven klad, verbeteren, volledig herschrijven, nakijken, ...)
Lennert De Geeter	Geen paginanummers (enkel brainstormen)	Brainstormen op papier/ digitaal op een tekentablet.
Roy De Volder	Geen paginanummers (enkel brainstormen)	Brainstormen op papier/ digitaal op een tekentablet.
Jorik Van Dorpe	Geen paginanummers (enkel brainstormen)	Brainstormen op papier/ digitaal op een tekentablet.
Lukas Verschraegen	Geen paginanummers (enkel brainstormen)	Brainstormen op papier/ digitaal op een tekentablet.

Lesweek 9: 27/02 - 5/03

Naam student	Paginnummers in rapport	Taak (bijv. schrijven klad, verbeteren, volledig herschrijven, nakijken, ...)
Lennert De Geeter	Pagina 7	Inleiding
Roy De Volder	Pagina 1,3	Hoofdblad opstellen
Jorik Van Dorpe	Pagina 1-11	Controle
Lukas Verschraegen	Pagina 1-11	Opstellen van het document volgens de regels uit het boek.

Lesweek 10: 5/03 - 12/03

Naam student	Paginnummers in rapport	Taak (bijv. schrijven klad, verbeteren, volledig herschrijven, nakijken, ...)
Lennert De Geeter	Pagina 23-25	Het schrijven van een kladversie.
Roy De Volder	Pagina 26-28	Het schrijven van een kladversie.
Jorik Van Dorpe	Pagina 22	Het schrijven van een kladversie.
Lukas Verschraegen	Pagina 13-21	Het schrijven van een kladversie.

Lesweek 11: 13/03 - 19/03

Naam student	Paginnummers in rapport	Taak (bijv. schrijven klad, verbeteren, volledig herschrijven, nakijken, ...)
Lennert De Geeter	Pagina 23-25	Het herschrijven naar een correcte versie.
Roy De Volder	Pagina 26-28	Het herschrijven naar een correcte versie.
Jorik Van Dorpe	Pagina 22	Het herschrijven naar een correcte versie.
Lukas Verschraegen	Pagina 13-21	Het herschrijven naar een correcte versie.

Lesweek 12: 20/03 - 26/03

Naam student	Paginnummers in rapport	Taak (bijv. schrijven klad, verbeteren, volledig herschrijven, nakijken, ...)
Lennert De Geeter	Pagina 13-19	Nakijken tekst.
Roy De Volder	Pagina 7-11 / 30	Controle van de lay-out.
Jorik Van Dorpe	Pagina 20-28	Nakijken tekst.
Lukas Verschraegen	Pagina 7-11 / 30	Opstellen van de lijsten volgens de regels uit het boek en verbeteren tekst.

Lesweek 13: 27/03 – 2/04

Naam student	Paginnummers in rapport	Taak (bijv. schrijven klad, verbeteren, volledig herschrijven, nakijken, ...)
Lennert De Geeter	Pagina 13-28	Verbeteren van fouten
Roy De Volder	Pagina 13-28	Verbeteren van fouten
Jorik Van Dorpe	Pagina 13-28	Verbeteren van fouten
Lukas Verschraegen	Pagina 13-28	Verbeteren van fouten

Lesweek 14: 3/04 – 9/04

Naam student	Paginnummers in rapport	Taak (bijv. schrijven klad, verbeteren, volledig herschrijven, nakijken, ...)
Lennert De Geeter	Pagina 7-11, 40-42	Lijsten bijwerken
Roy De Volder	Pagina 38	Schrijven van kladversie
Jorik Van Dorpe	Pagina 7-11, 40-42	Lijsten bijwerken
Lukas Verschraegen	Pagina 29-37	Schrijven van kladversie

Lesweek 15: 10/04 – 16/04

Naam student	Paginnummers in rapport	Taak (bijv. schrijven klad, verbeteren, volledig herschrijven, nakijken, ...)
Lennert De Geeter	Pagina 29-38	Nakijken
Roy De Volder	Pagina 38	Verder schrijven aan kladversie
Jorik Van Dorpe	Pagina 29-38	Nakijken
Lukas Verschraegen	Pagina 29-37	Verder schrijven aan kladversie

Lesweek 16: 17/04 – 23/04

Naam student	Paginnummers in rapport	Taak (bijv. schrijven klad, verbeteren, volledig herschrijven, nakijken, ...)
Lennert De Geeter	Pagina 29-38	Nakijken op fouten
Roy De Volder	Pagina 38	Herschrijven naar correcte versie
Jorik Van Dorpe	Pagina 29-38	Nakijken op fouten
Lukas Verschraegen	Pagina 29-37	Herschrijven naar correcte versie

Lesweek 17: 24/04 – 30/04

Naam student	Paginnummers in rapport	Taak (bijv. schrijven klad, verbeteren, volledig herschrijven, nakijken, ...)
Lennert De Geeter	Pagina 29-37	Nakijken tekst
Roy De Volder	Pagina 29-38	Nakijken op tekst en inhoud
Jorik Van Dorpe	Pagina 29-37	Nakijken op Inhoud
Lukas Verschraegen	Pagina 29-38	Controleren op fouten en lay-out in orde stellen

Lesweek 18: 1/05 – 7/05

Naam student	Paginnummers in rapport	Taak (bijv. schrijven klad, verbeteren, volledig herschrijven, nakijken, ...)
Lennert De Geeter	Pagina 35-36	Aanvullen van nieuwe ontwerpen
Roy De Volder	Pagina 31-42	Verbeteren van fouten in hoofdstuk 2
Jorik Van Dorpe	Pagina 45	Schrijven programmatie hoofdstuk 3
Lukas Verschraegen	Pagina 48	Intro schrijven voor hoofdstuk 4 van het verslag.

Lesweek 19: 8/05 – 14/05

Naam student	Paginnummers in rapport	Taak (bijv. schrijven klad, verbeteren, volledig herschrijven, nakijken, ...)
Lennert De Geeter	Pagina 37	Bespreking mechanische verwerking motoren
Roy De Volder	Pagina 43-44	Schrijven van bestukking
Jorik Van Dorpe	Pagina 45	Schrijven aanpassingen programmatie hoofdstuk 3
Lukas Verschraegen	Pagina 48	Het hoofdstuk connecties schrijven bij deel vier over testen

Lesweek 20: 15/05 – 21/05

Naam student	Paginnummers in rapport	Taak (bijv. schrijven klad, verbeteren, volledig herschrijven, nakijken, ...)
Lennert De Geeter	Pagina 54	Bespreking mechanische testen en verwerken van versies
Roy De Volder	Pagina 4	Schrijven van abstract
Jorik Van Dorpe	Pagina 54	Schrijven over de programmatie hoofdstuk 4
Lukas Verschraegen	Pagina 49	Het hoofdstuk werking schrijven bij deel vier over testen

Lesweek 21: 22/05 – 28/05

Naam student	Paginnummers in rapport	Taak (bijv. schrijven klad, verbeteren, volledig herschrijven, nakijken, ...)
Lennert De Geeter	Pagina 51	Bespreking van veiligheid binnen het testen van een prototype
Roy De Volder	Pagina 53	Schrijven van deel functionaliteit
Jorik Van Dorpe	Pagina 45	Aanpassen programmatie hoofdstuk 3 en over programmatie hoofdstuk 4
Lukas Verschraegen	Pagina 51	De conclusie formuleren

Lesweek 22: 29/05 – 4/06

Naam student	Paginnummers in rapport	Taak (bijv. schrijven klad, verbeteren, volledig herschrijven, nakijken, ...)
Lennert De Geeter	Alles	Opmaak en taalkundige controle
Roy De Volder	Alles	Controleren op fouten en opmaak
Jorik Van Dorpe	Pagina 45	Aanpassen programmatie hoofdstuk 3 en 4 en controleren op fouten
Lukas Verschraegen	Alles	Opmaak netjes plaatsen en helpen controleren op fouten

