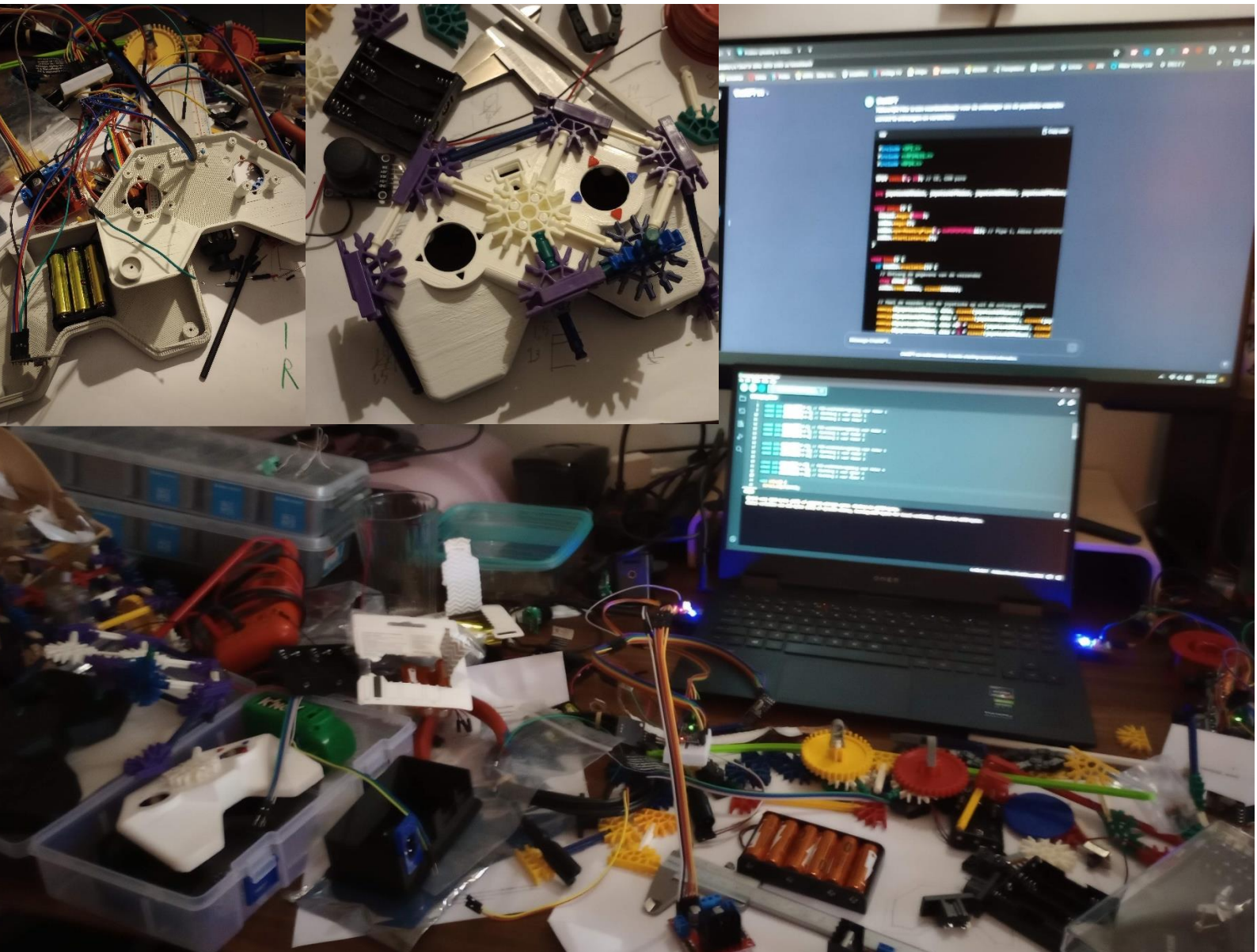


Hogeschool Utrecht

# Ontwerpen en productontwikkeling

Leraar Techniek



Florent Hofman (1765332)  
15-1-2024

## Inhoudsopgave

Inhoudsopgave .....	1
1. Oriëntatiefase .....	3
1.1. Dopaminebord booster .....	3
1.2. RC K'NEX motoren .....	3
1.3. Afweging .....	3
1.4. Voorlopige probleemstelling .....	4
1.5. Vooronderzoek .....	4
1.6. Reddit .....	5
1.7. Tekeningen .....	7
2. Analyse fase .....	9
2.1. Plan van aanpak .....	9
2.2. Plan van eisen en wensen .....	9
2.2.1. Eisen .....	9
2.2.2. Wensen .....	9
2.3. Hoofdfuncties en deelfuncties .....	9
2.3.1. Hoofdfunctie .....	9
2.3.2. Deelfuncties .....	10
2.4. Onderzoek .....	11
2.5. Doelgroeponderzoek .....	11
2.6. Marktonderzoek .....	12
2.6.1. Conclusie marktonderzoek 1 .....	14
2.7. Knex overige onderdelen ideeën .....	14
3. Idee fase .....	15
3.1. Schetsen .....	15
3.2. Vorm knex .....	21
3.3. favoriet .....	22
4. Conceptfase .....	23
4.1. Onderdelen lijst .....	23
4.2. Pcb ontwerpen .....	24
4.3. Onderzoekje afmetingen Joystick .....	24
5. Materialisatiefase .....	25
5.1. Arduino code .....	28
5.1.1. Controller .....	28
5.1.2. Ontvanger .....	30
6. Productievoorbereiding .....	34

7.	Evaluatie en testfase .....	35
8.	Definitiefase.....	36
9.	Didactische koppeling.....	37
9.1.	Projecten .....	37
9.2.	Fushion360.....	37

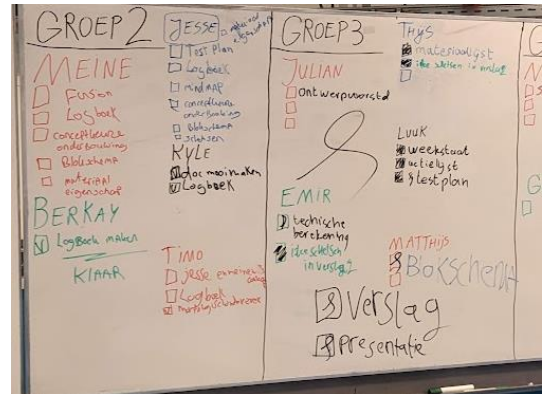
## 1. Oriëntatiefase

Er zijn 2 ideeën waar ik aan zit te denken om uit te gaan werken. Uiteindelijk wordt het er 1 maar ik weet niet goed welke ik moet gaan kiezen. Ik ga ze beide even omschrijven.

### 1.1. Dopaminebord booster

In mijn projectlessen maak ik vaak gebruik van een takenbord. Op dit bord moeten de studenten hun taken zetten. Als ze met een taak klaar zijn dan mogen ze hem afvinken en kunnen ze weer door. Het idee is dat als je iets af vinkt je dopamine krijgt.

Nu wil ik deze dopamine, vergroten. Door de afvink box van ledjes te maken. Als ze dus een taak afhebben dan zullen de ledjes gaan knipperen, en een klein showtje geven. Waarbij ik dus hoop dat het meer dopamine zal geven en dat de studenten dus extra gemotiveerd worden om een taak af te willen ronden.



### 1.2. RC K'NEX motoren

Vroeger heb ik heel veel met de K'NEX gespeeld. Soms gewoon gebouwd wat er op de instructie stond maar soms ook gewoon eigen ontwerp. Dit werden uiteindelijk steeds technischere en complexere bouwwerken. Tot ik een keer graag een bestuurbare auto wilde maken. Dit ging niet zomaar, ik weet dat ik een keer wat geprobeerd heb. Maar het liefst zou ik gewoon een mooie afstandsbediening willen die motoren harder en zachter kan zetten. En een Servomotor om bijvoorbeeld te kunnen sturen. Zodat ik een mooie bestuurbare auto kan maken van de K'NEX en uiteindelijk mijn kleine neefjes er ook een kunnen bouwen.



### 1.3. Afweging

Eerst was ik erg in twijfel voor welk idee ik zou gaan. De dopaminebord booster is een uniek idee (voor zover ik weet). RC K'NEX is niet zozeer uniek in de zin van dat het erg lijkt op iets wat al bestaat. Bestuurbare technisch lego bijvoorbeeld. Maar voor zover mijn onderzoek nu gelopen heeft lijkt het niet voor de K'NEX te bestaan. Wel een K'NEX drone die je vast kan maken aan je bouwwerken, maar dat is niet wat ik graag wil.

Voor nu ga ik mij toch focussen op de RC K'NEX. Dit omdat de dopamine booster toch veel dingen aan zitten wat het toch tot een flop kan laten lopen. Zoals: Het kan gezien gaan worden als kinderachtig, het kan misschien helemaal geen extra dopamine geven, misschien wel minder. Als het niet werkt dan zijn de studenten gelijk niet meer vooruit te branden. Het moet na mij idee een te perfect en te geavanceerd uit gewerkt worden om het echt goed te laten werken. Dit is dan waarschijnlijk ook niet de moeite voor wat het zou gaan opleveren. Voor de RC K'NEX is het simpel en weinig externe factoren die het tot een flop kunnen laten lopen.

#### 1.4. Voorlopige probleemstelling

Mijn gemis aan verdieping en verbreding van het bouwspeelgoed K'NEX. Met name van het draadloos besturen van je bouwwerken/ voertuigen.

#### 1.5. Vooronderzoek

Ik heb nog extra gezocht of ik niet iets wilde gaan maken wat toch al bestond. Het enige wat ik ben tegengekomen is deze drone. Maar die bevalt mij niet, het is te simpel. Je gaat ook geen drone boven op je hijskraan zetten om deze te laten hijsen.

[https://www.K'NEXusergroup.org.uk/en/image\\_K44421model2.html](https://www.K'NEXusergroup.org.uk/en/image_K44421model2.html)

Op een website van K'NEX gebruikers staan ook alle motoren die er zijn op een rij.



[https://www.knexusergroup.org.uk/en/info\\_K92755.html](https://www.knexusergroup.org.uk/en/info_K92755.html)

Deze motoren met besturing, geven een idee wat ik zou willen. Alleen dan draadloos.

Op deze zelfde website vond ik deze motor



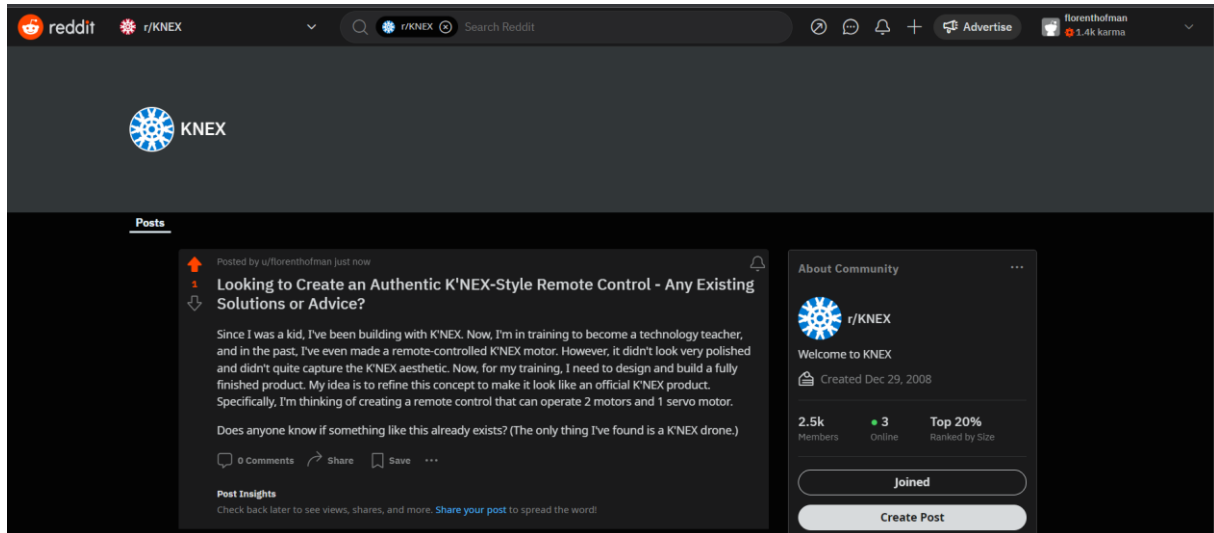
Als je deze samen klikt dan kan je hem besturen. Stel dit zou draadloos zijn dan zou dit ook al een goede oplossing zijn voor mijn probleem. Dit geeft in ieder geval wel inspiratie van hoe ik mijn ontwerp er uit kan gaan laten zien.





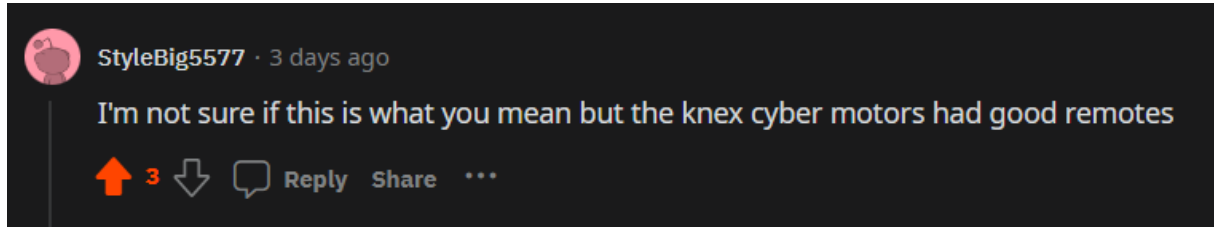
## 1.6. Reddit

Ook heb ik nog een vraag uit gesteld op Reddit R/KNEX. Om hier nog te vragen of dit niet al bestaat. Misschien dat ik wat over het hoofd gezien heb. De tekst heb ik in het Nederlands getypt in ChatGPT <https://chat.openai.com/share/950eceb1-3f52-4fd3-a0cf-88f59133971d> en gevraagd of die het kon vertalen en het goed kon laten lopen. Zodat het ook wat aandacht trekt naar de juiste Redditors.



[https://www.reddit.com/r/K'NEX/comments/16d9zxv/looking\\_to\\_create\\_an\\_authentic\\_KNEXstyle\\_remote/](https://www.reddit.com/r/K'NEX/comments/16d9zxv/looking_to_create_an_authentic_KNEXstyle_remote/)

Op deze vraag kwam weinig respons. Maar wel 1 bericht die mij wel tot een product heeft geleid die ik nog niet kende. Die mogelijk de oplossing kan zijn voor ons probleem.



Nu heb ik deze opgezocht en kwam ik bij dit product uit.



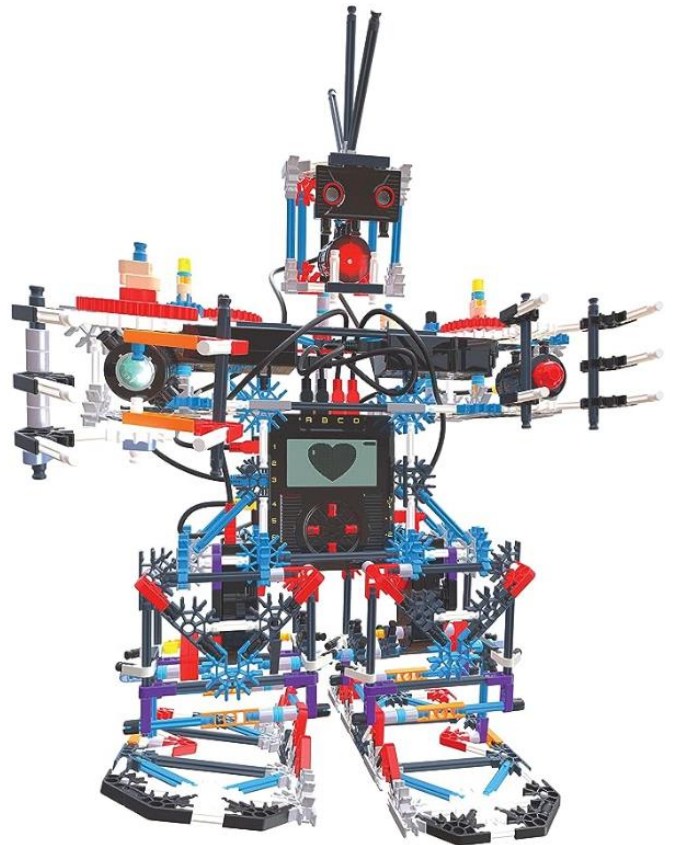
Het draadloos besturen van de motoren is bij deze set mogelijk. Ook kun je deze set zelf programmeren doormiddel van programmeer tokens. Hoe dit exact werkt heb ik nog geen beeld bij.

Deze vondst weerhoudt mij er niet van om toch verder te gaan ontwerpen. Omdat dit product een oud product is en het er allemaal vrij complex uit ziet. Mij eindproduct ga ik dan ook als eis stellen dat die eenvoudig in gebruik is en dat die er ook simpel uitziet. Zodat een kind van 5 hem kan gebruiken en toepassen.

Tijdens zoeken kwam ik ook deze nog tegen: KNEX robotics

<https://www.amazon.com/KNEX-Education-Robotics-Building-Engineering/dp/B01B55C6PA?tag=takweb-20>

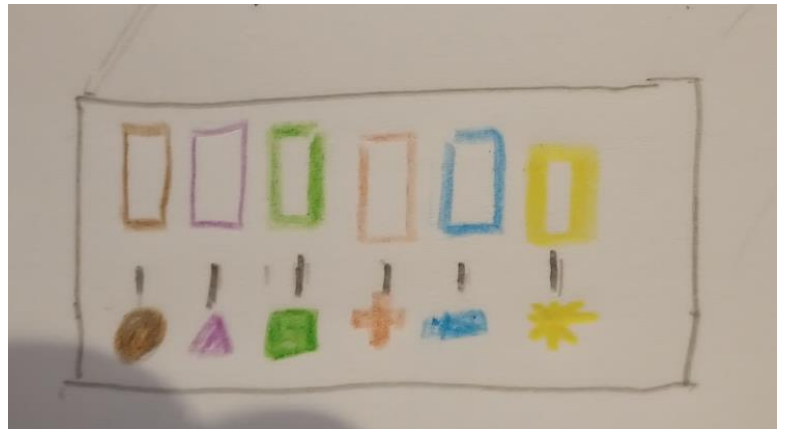
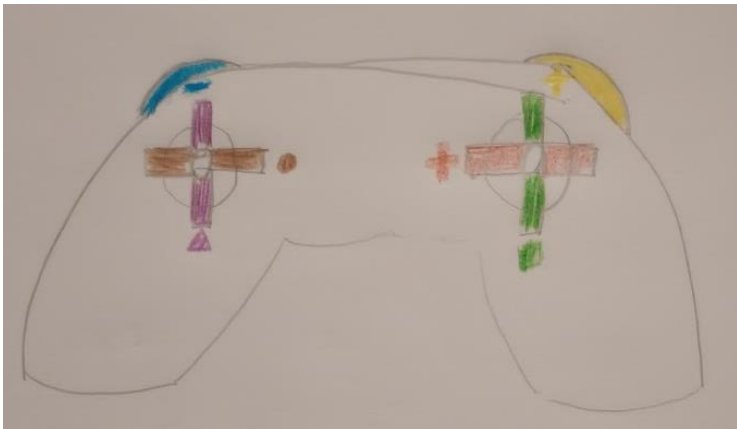
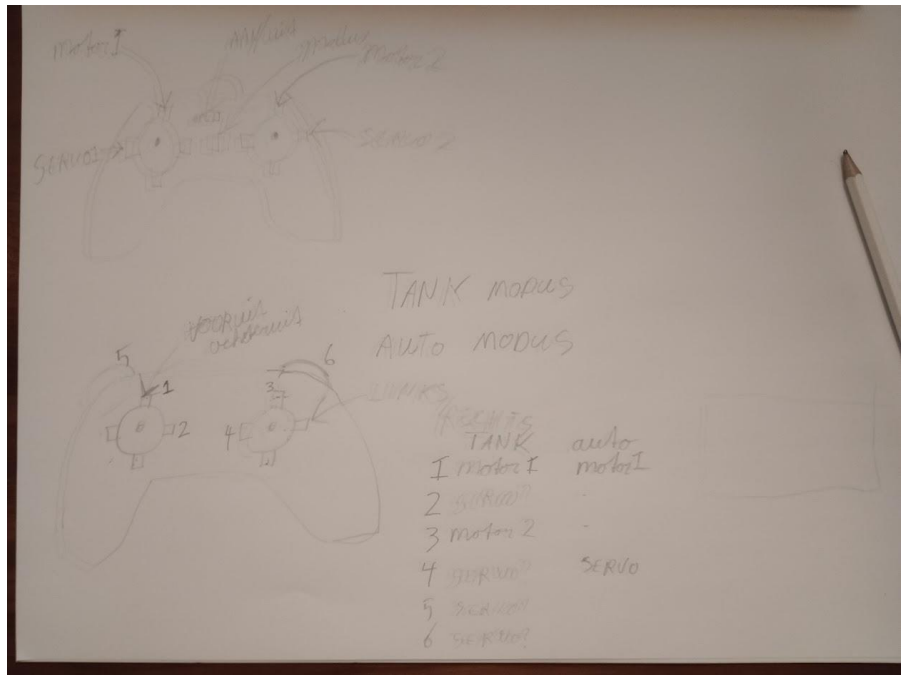
Ook deze is vrij complex en voldoet dus niet aan de eisen.



## 1.7. Tekeningen

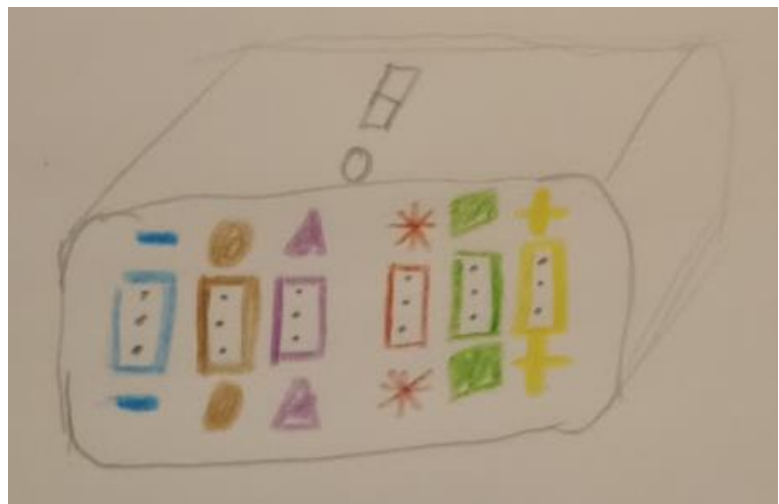
Tijdens het tekenen kwam ik op ideeën. Om het tot een goed en simpel product te maken. Graag wilde ik de afstandsbediening op verschillende manieren kunnen gebruiken. Maar met het uitwerken hiervan werd het steeds ingewikkelder.

Waardoor ik uiteindelijk op het idee kwam om met kleuren te gaan werken, en daar de vrijheid in te houden. Waardoor het voor de gebruiker alsnog simpel blijft.



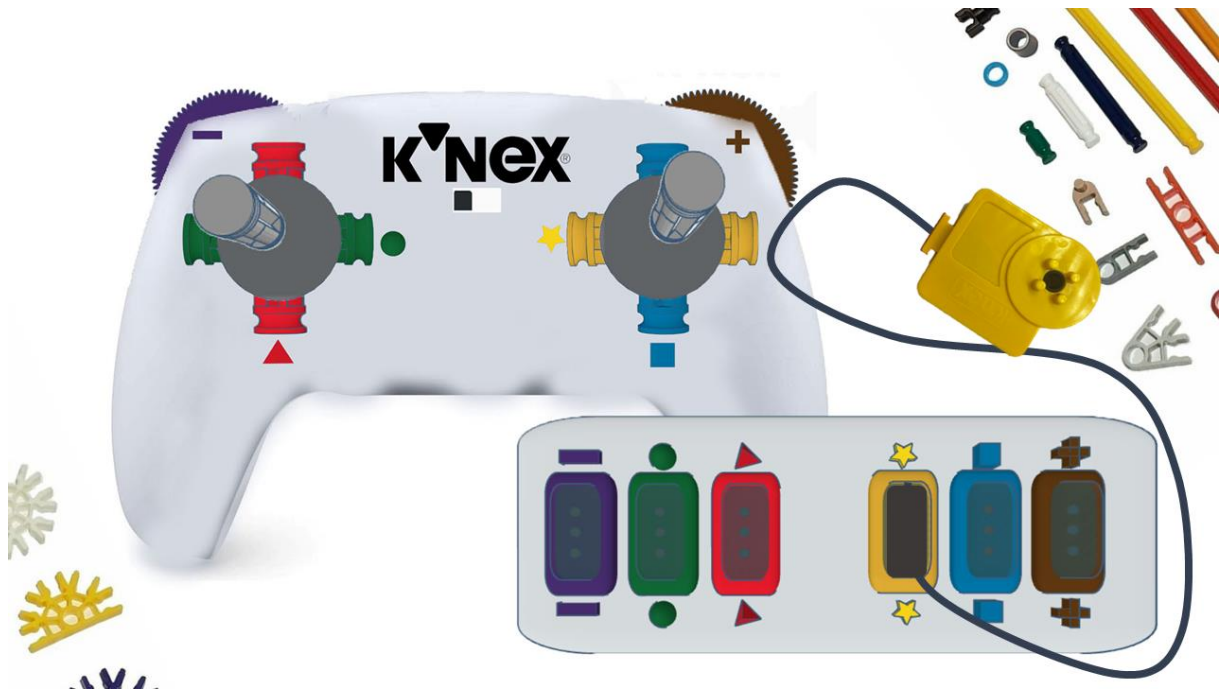
Ook kwam ik hier op het idee om figuurtjes toe te voegen zodat mensen die kleurenblind zijn er ook goed mee overweg kunnen.

Ook heb ik ervoor gekozen om de verdeling op de ontvanger wat te veranderen. Zodat het sneller duidelijk is wat waarbij hoort. Drie linkerkant van de controller en 3 aan de rechterkant.





Uiteindelijk heb ik nog in Photoshop en Tinkercad een ontwerp gemaakt van hoe ik hem er graag uit zou willen laten zien.



Hierbij heb ik ook de kleuren gewijzigd zodat er uit de kleuren ook al valt te halen waarbij het hoort. Zo horen rood en blauw bij elkaar, en Groen en geel. Of de volgorde nu helemaal naar mijn zin is. Ben ik nog niet uit.

## 2. Analyse fase

### 2.1. Plan van aanpak

Hoe ga ik het aanpakken? Voornamelijk door veel te proberen en te testen. Een tekening maken en deze uit printen. Kijken of het klopt past en zo niet weer opnieuw.

### 2.2. Plan van eisen en wensen

#### 2.2.1. Eisen

- K'NEX stijl
- K'nex compatible
- Makkelijk in gebruik, kind van 6 eisen regelgeving
- Makkelijk te begrijpen, test hoe kind van 6 er mee omgaat
- Minimaal 2 motors
- Minimaal 1 servo
- Draadloos signaal van controller naar ontvanger
- Werkt op batterijen of accu
- 2 joysticks
- Minimaal 4 signalen uit kunnen sturen
- Afstand van minimaal 20 meter kunnen besturen, zonder obstructies
- Minimaal 8 sets moeten gebruikt kunnen worden in het nabij zijn van elkaar. (unieke id's)
- Minimaal 2 uur meegaan.
- Aan en uitknop, controller en ontvanger
- Led indicatie van aan of uit.
- Led indicatie van Verbonden zijn.
- Batterijen of accu moeten vervangen kunnen worden door gebruik of ouder 18+
- Moet door een 5 jarige te gebruiken zijn
- Moet aan EU richtlijnen voldoen
- 

#### 2.2.2. Wensen

- Doorzichtige elementen er in. Zodat je kan zien wat er in de behuizing gebeurt, tandwielen bijvoorbeeld.
- Kleuraanduiding van ontvanger overeen laten komen met controller joysticks. (assen)
- Oplaadbaar
- Uitbreid mogelijkheid voor eventueel uitbreidingskit om te kunnen programmeren, sensoren ed.
- Batterij indicatie hebben
- Uitbreid mogelijkheid voor controller, Display

Deze eisen en wensen heb ik uiteindelijk bijgesteld omdat dit te groot zou worden en te complex. Uiteindelijk ben ik voor 4 motoren gegaan in plaats van 6. Bij welk project zou je nou ook 6 motoren gebruiken? 4 is al veel. Ook heb ik de servo's er uit gehaald omdat dit dan nog een extra ontwerp zou moeten worden. Nu moet er gewoon creatief gebouwd gaan worden met de motoren die we nu hebben.

### 2.3. Hoofdfuncties en deelfuncties

#### 2.3.1. Hoofdfunctie

**Aandrijving:** De motor kan worden gebruikt om het K'NEX-voertuig of -apparaat aan te drijven.

**Sturing:** Motoren kunnen worden gebruikt om stuurmechanismen te bedienen, waardoor je het apparaat in verschillende richtingen kunt bewegen.

**Specifieke Bewegingen:** Je kunt de motoren gebruiken om specifieke bewegingen uit te voeren, afhankelijk van je ontwerp, zoals het optillen van delen of het draaien van componenten.

### 2.3.2. Deelfuncties

**Snelheidsregeling:** Mogelijk kun je de snelheid van de motoren aanpassen voor verschillende doeleinden.

**Richtingscontrole:** Als je de motoren in beide richtingen kunt laten draaien, biedt dit meer flexibiliteit in bewegingen.

**Aanpassingen:** Je kunt extra functies hebben toegevoegd, zoals lichten, geluiden of andere bewegende delen, afhankelijk van je creativiteit.

## 2.4. Onderzoek

- Joystick afmetingen
- Voeding voor Arduino pro mini

Deze onderzoeken heb ik gedaan bij het tinkeren.

## 2.5. Doelgroeponderzoek

### Doelgroepen collage

Graag zou ik het product zo simpel houden dat een 5 jarige het zelf kan gebruiken. Maar ook dat een volwassene van 18 of ouder er. Prettig mee overweg kan. Maar het is beter om te focussen op een kleine specifieke groep. Om zo tot een goed product te komen. Nu is 5 tm 18 een grote groep. Ik denk dat ik voor de doelgroep ervoor kies om er tussen in te gaan zitten. leeftijd 12 jaar. Met advies 5+



## 2.6. Marktonderzoek

Zo doet lego het dus. Extreem simpel. En helemaal in de Lego stijl. Precies zo zou ik het ook willen alleen dan voor de K'nex.

[https://www.bol.com/nl/nl/p/lego-powered-up-afstandsbediening-88010/9300000090157085/?Referrer=ADVNLGOO002027-S--9300000090157085&gclid=Cj0KCQjw9fqkBhDSARIsAHlcQYS2fEPqiQls4YbnHaYUBQCpptsLWFNgmVHKGIEicTgdLb1sS0WwHVUaAtgREALw\\_wcB](https://www.bol.com/nl/nl/p/lego-powered-up-afstandsbediening-88010/9300000090157085/?Referrer=ADVNLGOO002027-S--9300000090157085&gclid=Cj0KCQjw9fqkBhDSARIsAHlcQYS2fEPqiQls4YbnHaYUBQCpptsLWFNgmVHKGIEicTgdLb1sS0WwHVUaAtgREALw_wcB)





Ook lego alleen dan iets complexere look.

[https://www.bol.com/nl/nl/p/dw4trading-afstandbestuurbare-motor-met-receiver-controller-en-batterijhouder-technics-compatibel-met-grote-merken/9300000025851461/?bltgh=rgEnrkGPdjfhmeHGDdhLew.2\\_27.30.ProductImage](https://www.bol.com/nl/nl/p/dw4trading-afstandbestuurbare-motor-met-receiver-controller-en-batterijhouder-technics-compatibel-met-grote-merken/9300000025851461/?bltgh=rgEnrkGPdjfhmeHGDdhLew.2_27.30.ProductImage)

# Speed controlled



[https://www.amazon.nl/LEGO-IR-Remote-Control-8885/dp/B003FO2FDS?source=ps-sl-shoppingads-lpcontext&ref\\_=fplfs&psc=1&smid=AAPBTTBBQGLZ](https://www.amazon.nl/LEGO-IR-Remote-Control-8885/dp/B003FO2FDS?source=ps-sl-shoppingads-lpcontext&ref_=fplfs&psc=1&smid=AAPBTTBBQGLZ)

#### 2.6.1. Conclusie marktonderzoek 1

Lego heeft voor elk bouwpakket wat ze hebben bijna een andere controller. Deze 3 verschillen al een hoop van elkaar. Het is dan ook allemaal voor een andere toepassing. Wat mijn idee was, was het maken van een controller met ontvanger voor alle toepassingen. Dit wil ik nu nog wel proberen door te zetten maar ik merk wel dat het een gevecht wordt met het een simpel product te laten zijn.

#### 2.7. Knex overige onderdelen ideeën



<https://knex.parts/product/sound-unit-blue/?v=796834e7a283>

Voor de ontvanger wil ik ook zulke pinnen op de case net als bij dit onderdeel van de knex.

### 3. Idee fase

#### 3.1. Schetsen

Met het schetsen kwam ik al tot de eerste problemen, en ook oplossingen. Maar niet direct een goede oplossing.

Met het schetsen de controllers opgezocht wat is standaard voor een auto. Daar kwam ik deze afbeelding bij tegen. Ik was al aan het denken aan modules voor verschillende manieren van

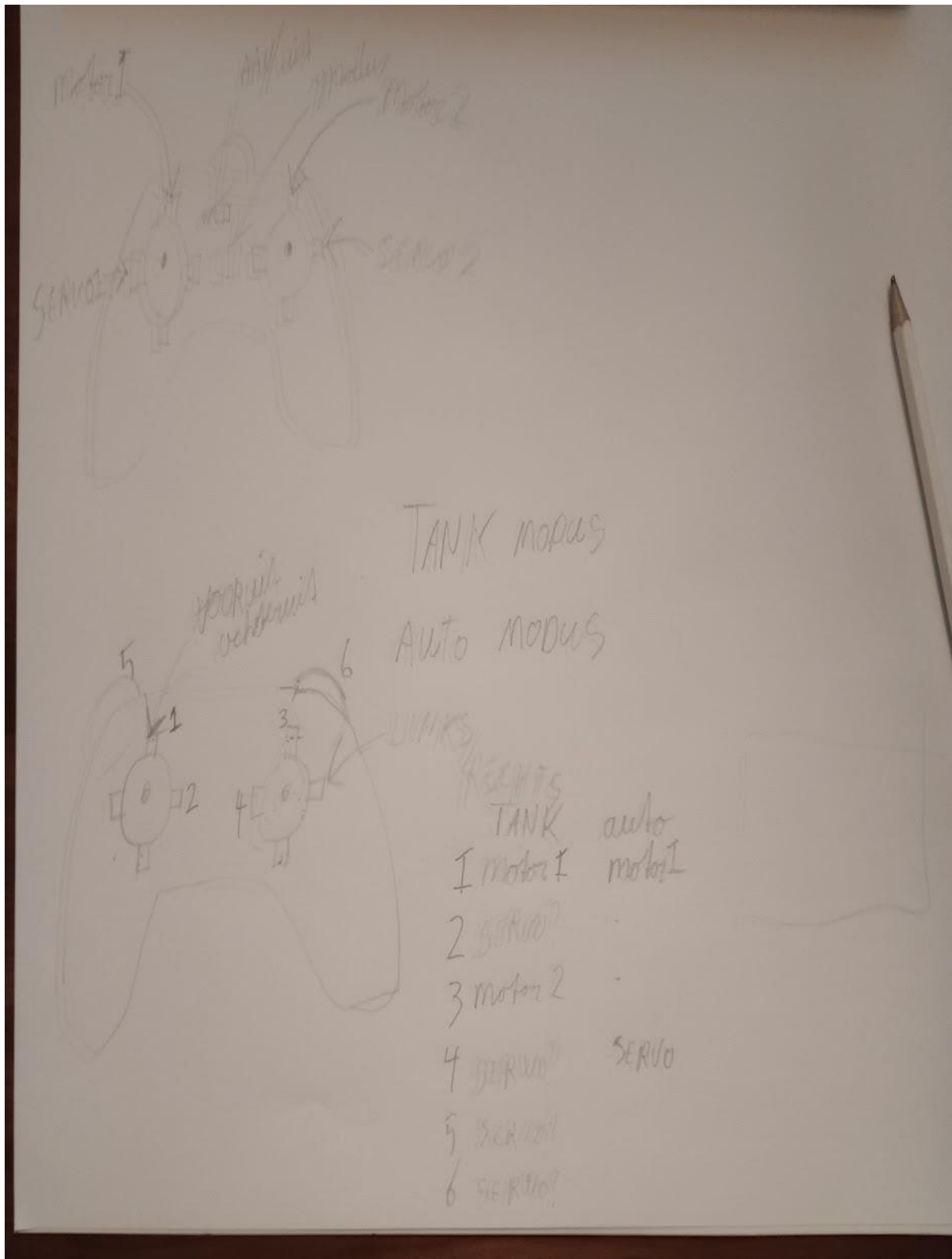
aansturen. Dus 2 knoppen. Te overzien. Maar toen bedacht ik mij wat als iedereen deze zo heeft. Dan wil ik dat iedereen zijn eigen motoren kan bedienen en elkaar daarmee dus niet in de weg zit. Misschien moet hier dus ook nog een knop voor. Maar dat maakt hem dan wel ingewikkeld. Maar misschien is het ook softwarematig wel op te lossen. Zodat de klant hier geen omkijken naar heeft.



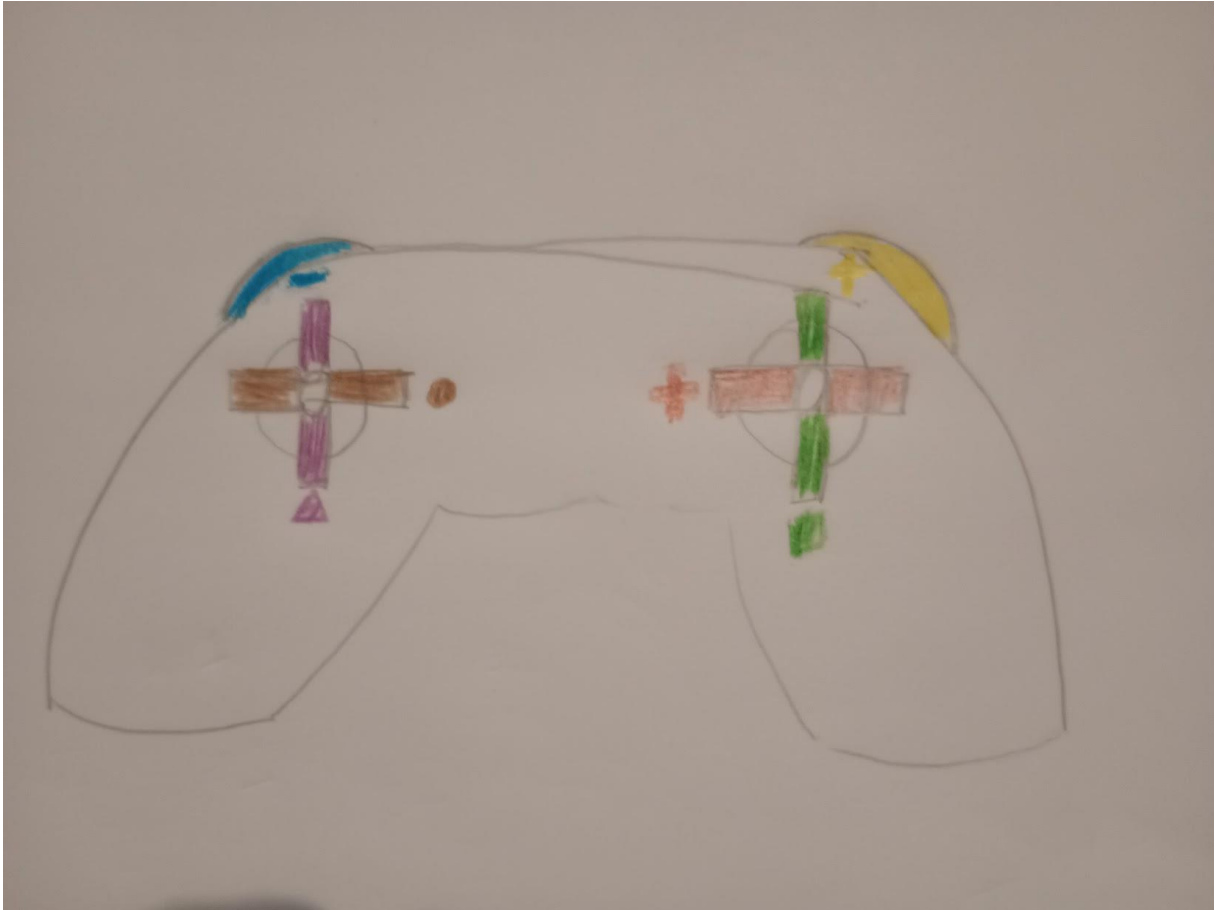
De toewijzing van de controller naar de motoren. Is erg ingewikkeld. Ik heb het idee dat dit is waar de ontwerper van de Cyber KNEX ook tegenaan is gelopen en toen gekozen heeft voor de vrijheid voor de klant. Maar het daarmee dus wel ingewikkeld heeft gemaakt.

Ook merk ik dat ik eigenlijk beter nog een marktonderzoek kan doen. Zodat ik weet hoe ze deze problemen bij bijvoorbeeld lego hebben opgelost.

<https://nl.aliexpress.com/item/1005004921561462.html>



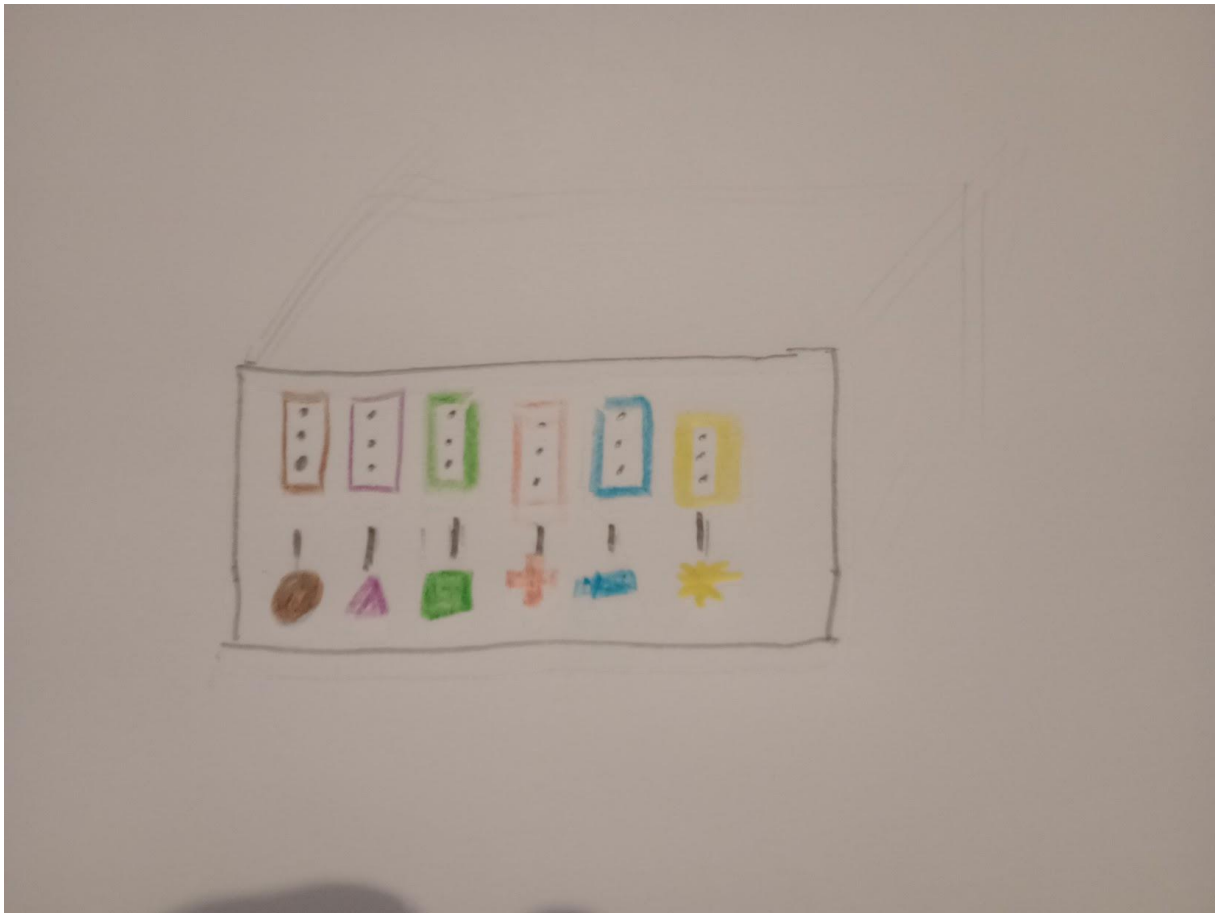
Hoe de controller er uit gaat zien hangt dus ook erg af van hoe de ontvanger er uit gaat zien. Qua functies.



Idee is om kleuren aan de controller te matchen met de ontvanger. En dat de gebruiker dan de kabel van zijn motor kan doen in de kleur (controller naar wens)

Oplossing: De kleuren bij de controller per knop/as, komen overeen bij de ontvanger met aansluitpunten. De gebruiker is er dan ook vrij in om wat die dan ook maar wilt, aan te sluiten op de ontvanger.



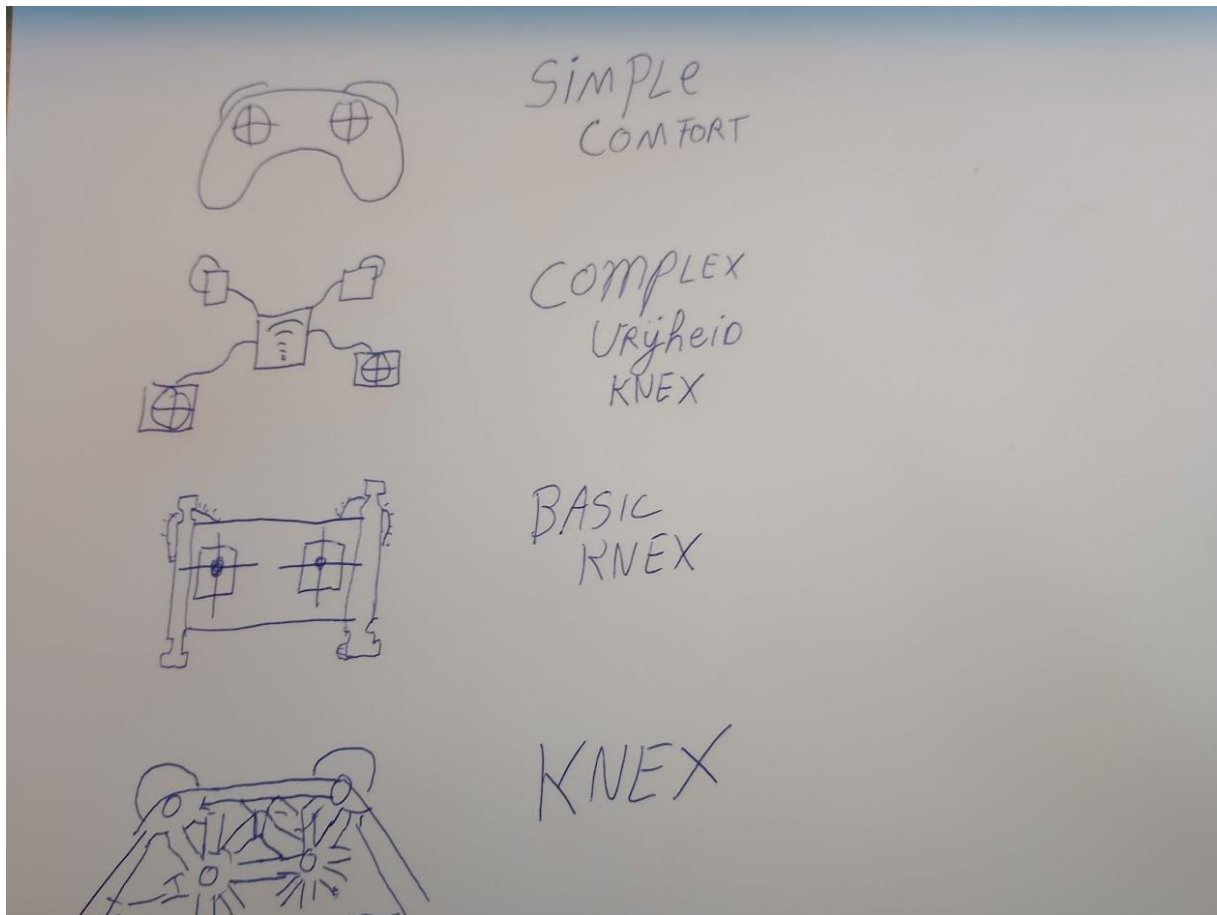


Met het tekenen van de kleuren, kwam ik er achter dat het ook beter is om de volgorde op de ontvanger enigszins ook overeen te laten komen met de controller. Misschien is het ook beter om dit in 2 groepen te doen. Linker kant en rechter kant. Ook is het dan beter om de symbolen er boven te doen. Zodat je het nog kan zien als je er een kabel in hebt. Alhoewel misschien is het beter aan beide zijde aangezien we niet weten wat de positie gaat worden in het bouwwerk.

Ook is het hier nog goed om over de kleuren na te denken. Zoals rood hoort bij blauw. Dus die gebruiken voor de verticale as. En bijvoorbeeld geel en groen bij de horizontale as. Waarbij dan paars en bruin voor de overige. Deze zijn een stuk minder logisch, hier kan nog over nagedacht worden.



Probleem: De kant van de motor, welke kant die op draaien. Wil ik graag gewoon dat de kabel omgedraaid zou willen. Of een kind dit gaat begrijpen is nog de vraag. Voor mij is dit logisch en een directe oplossing. Voor mijn vriendin al niet. Dus laat staan dat een kind dit al direct zou verzinnen. Hier moet dus een oplossing op bedacht worden. Of net zoals lego het doet, een knop erbij met een symbool erbij die dit aangeeft. Of een symbool bij de ontvanger om de kabel om te draaien. Maar hoe exact en of dit te begrijpen is. Is nog de vraag. Dit moet getest / onderzocht worden. Maar dat is wel waar ik voor wil gaan, om het product zo simpel mogelijk te laten zijn. Als we voor de knop erbij gaan dan zou dat betekenen dat we 6 knoppen moeten toevoegen. Terwijl het dus ook zonder kan.

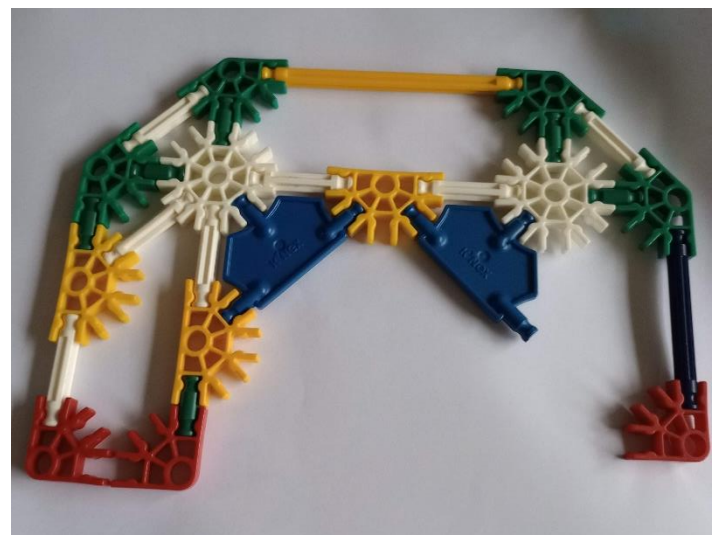
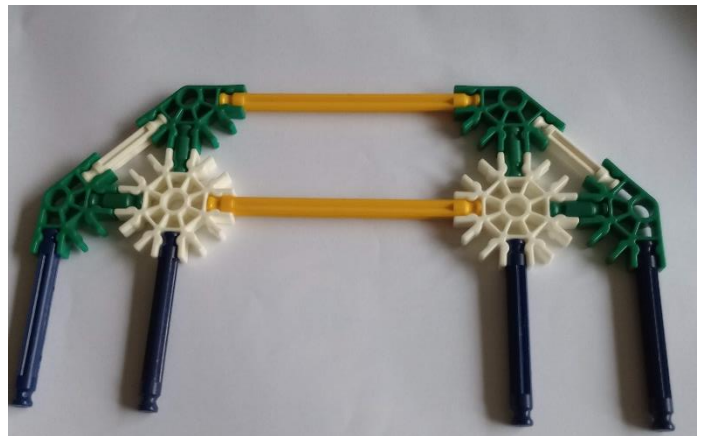
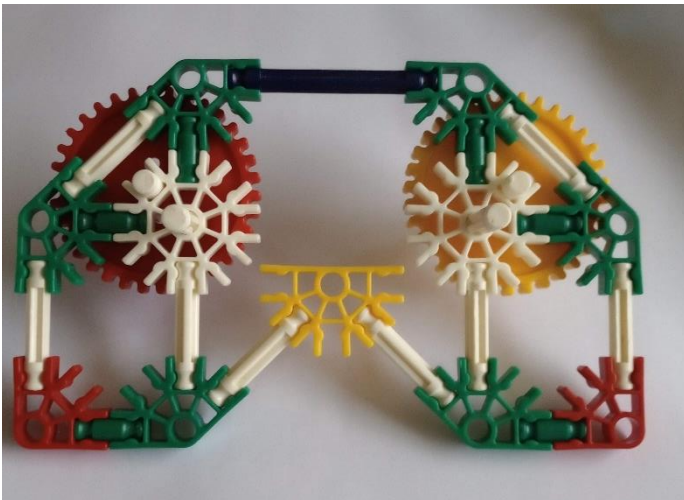
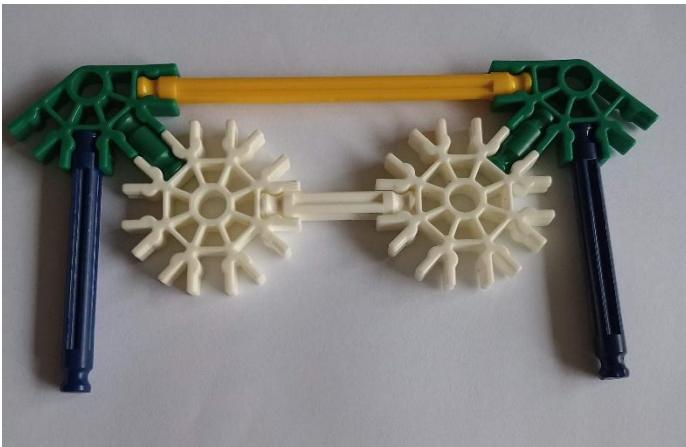


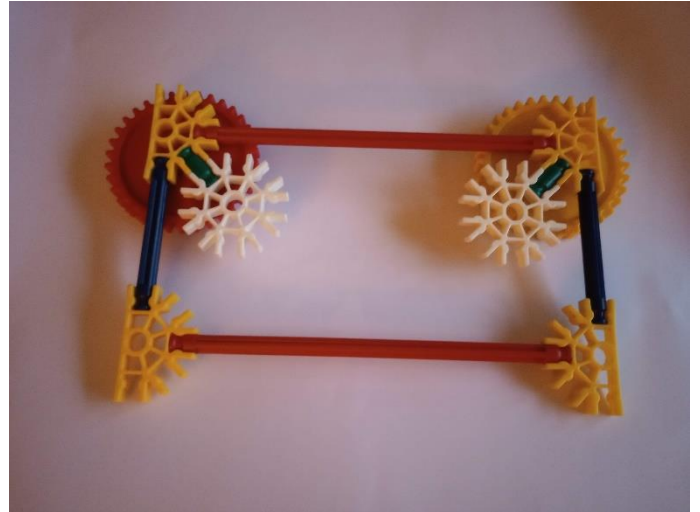
Ook heb ik nog 4 verschillende controllers getekend om zo ook iets meer op nieuwe ideeën uit te komen.

### 3.2. Vorm knex

Om op ideeën te komen heb ik verschillende vormen van controllers gemaakt vanuit K'nex.

Uit deze proef komen al 2 ergonomische verschillen. Ik was vrij groot begonnen aangezien ik grote handen heb. Maar de doelgroep heeft ook kleine handen. Mijn bevinding was dat ik met grote handen ook een kleine controller kan gebruiken. Maar andersom niet. Voor deze ergonomische afmetingen kan ik een informatie halen van deze website: <https://dined.io.tudelft.nl/en> mijn collega wees mij hierop. Floris Verstraelen

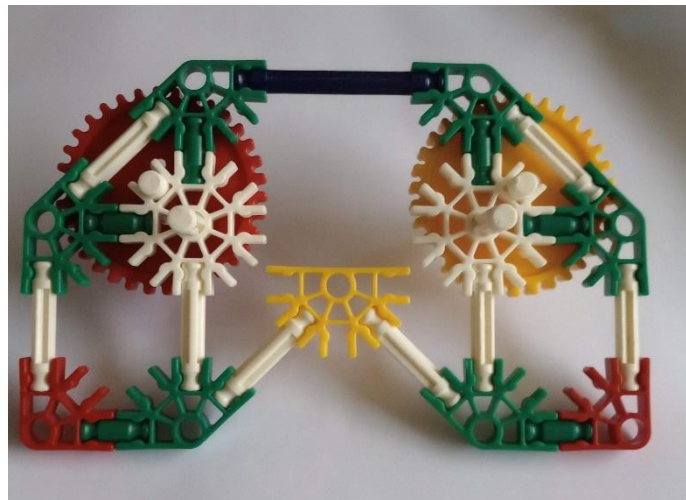




### 3.3. favoriet

Deze heeft uiteindelijk de vorm die het meest prettig in de hand ligt. Het enige nadeel wat hier nog op kan treden is dat de batterijen en elektronica er niet gemakkelijk in gaan passen.

Op basis van dit model ga ik in Fushion360 een ontwerp tekenen.





## 4. Conceptfase

### 4.1. Onderdelen lijst

aantal	naam	Waar
2	Arduino nano	<a href="https://www.tinytronics.nl/shop/nl/development-boards/microcontroller-boards/arduino-compatible/nano-v3.0-usb-c-compatible-losse-headers">https://www.tinytronics.nl/shop/nl/development-boards/microcontroller-boards/arduino-compatible/nano-v3.0-usb-c-compatible-losse-headers</a>
2	joystick	<a href="https://www.tinytronics.nl/shop/nl/schakelaars/manuele-schakelaars/joysticks/ps2-joystick">https://www.tinytronics.nl/shop/nl/schakelaars/manuele-schakelaars/joysticks/ps2-joystick</a>
2	Potmeter met terugslag	<a href="https://www.tinytronics.nl/shop/nl/componenten/weerstanden/potmeters/10k%CF%89-duimwiel-potmeter-2-kanaals-lineair-1">https://www.tinytronics.nl/shop/nl/componenten/weerstanden/potmeters/10k%CF%89-duimwiel-potmeter-2-kanaals-lineair-1</a>
2	RF module NRF24L01	<a href="https://www.tinytronics.nl/shop/nl/communicatie-en-signalen/draadloos/rf/modules/nrf24l01-wireless-module-zwart">https://www.tinytronics.nl/shop/nl/communicatie-en-signalen/draadloos/rf/modules/nrf24l01-wireless-module-zwart</a>
2	Aan uit knop	<a href="https://www.tinytronics.nl/shop/nl/schakelaars/manuele-schakelaars/schuifschakelaars/inbouw-schakelaar-on-off-ss22g08">https://www.tinytronics.nl/shop/nl/schakelaars/manuele-schakelaars/schuifschakelaars/inbouw-schakelaar-on-off-ss22g08</a>
2	Pushbutton voor connecten	<a href="https://www.tinytronics.nl/shop/nl/schakelaars/manuele-schakelaars/printplaat-schakelaars/breadboard-tactile-pushbutton-switch-momentary-2pin-6*6*5mm">https://www.tinytronics.nl/shop/nl/schakelaars/manuele-schakelaars/printplaat-schakelaars/breadboard-tactile-pushbutton-switch-momentary-2pin-6*6*5mm</a>
x	Leds	
2	Baterijhouder (wat voor?)	
6	Motor driver(wat voor?)	<a href="https://www.tinytronics.nl/shop/nl/mechanica-en-actuatoren/motoraansturingen-en-drivers/dc-motoraansturingen-en-drivers/l293d-motordriver-16-pin-dip">https://www.tinytronics.nl/shop/nl/mechanica-en-actuatoren/motoraansturingen-en-drivers/dc-motoraansturingen-en-drivers/l293d-motordriver-16-pin-dip</a>
5	Motor driver	<a href="https://www.tinytronics.nl/shop/nl/mechanica-en-actuatoren/motoraansturingen-en-drivers/dc-motoraansturingen-en-drivers/l293d-motor-shield-met-shiftregister">https://www.tinytronics.nl/shop/nl/mechanica-en-actuatoren/motoraansturingen-en-drivers/dc-motoraansturingen-en-drivers/l293d-motor-shield-met-shiftregister</a>

#### 4.2. Pcb ontwerpen

Ik heb nog gedacht om een pcb te gaan ontwerpen. Hiervoor heb ik een test gemaakt in Fushion360 om hiermee te leren werken. Hier zijn nog overige websites waaraan in gedacht heb om er 1 te ontwerpen.

Kicad.com

Easyeda.com

<https://www.snapeda.com/search/?q=joystick&search-type=parts>

chrome-extension://efaidnbmnnnibpcajpcgicfindmkaj/https://www.ti.com/lit/ds/symlink/l293.pdf

fushion360

#### 4.3. Onderzoekje afmetingen Joystick

<https://www.electronicoscaldas.com/datasheet/802-30110-A.pdf>

<https://www.thingiverse.com/thing:6207307>

## 5. Materialisatiefase

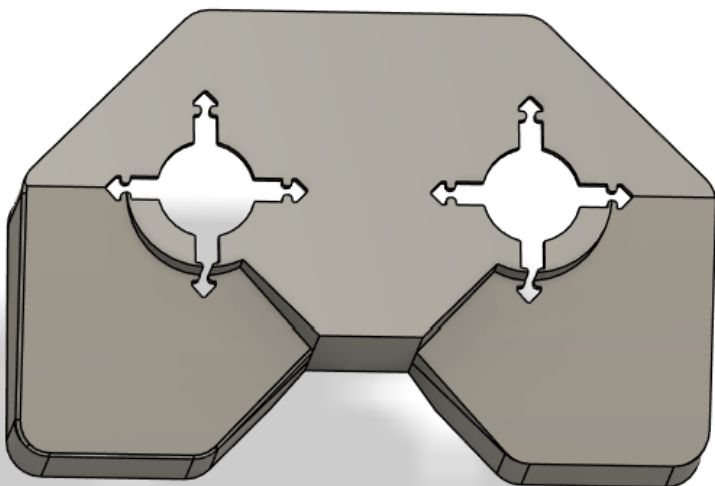
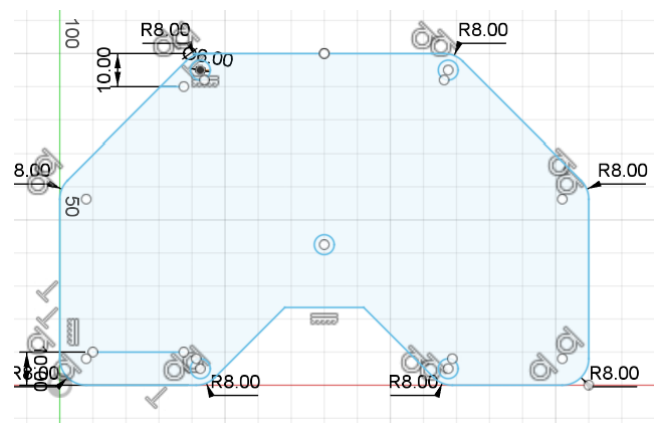
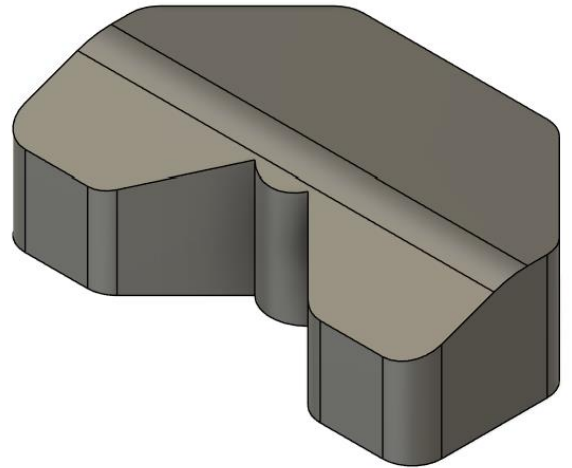
Ik heb ervoor gekozen om met Fusion360 te gaan tekenen omdat dit op mijn werk ook gebruikt wordt en les in gegeven. Regelmatig krijg ik dan ook vragen over dit programma. Maar omdat ik er zelf nog nooit echt mee gewerkt heb. Kan ik hier vaak geen antwoord op geven.

De eerste tekening in fusion360. Vooral gebaseerd op de knex vorm die ik had gemaakt. Nu had ik bij deze eerste tekening nog niet goed met de afmetingen rekening gehouden en ben ik gewoon gestart. Hierdoor kreeg ik wel al de basisvaardigheden binnen Fusion en ontdekte ik wat er werkt en wat niet.

Vervolgens ben ik meer naar de maten gaan kijken en heb ik de vorm die ik met knex had gemaakt nagetekend op schaal waarna ik verder ben gaan tekenen.

Uiteindelijk kwam er een ontwerp uit wat ik heb geprint. Voor de pijltjes om aan te geven met kleuren had ik een uitsparing gemaakt en voor die vorm heb ik mij laten inspireren door het uiteinde van een knex staaf.

Dit ontwerp had nog veel scherpe hoeken en ook de pijltjes waren eigenlijk niet helemaal naar mijn zin. Dit komt waarschijnlijk omdat het een volledig eigen ontwerp is. Later heb ik dit nog aangepast. Ook kwam hij vrij groot en lomp over. Ook kreeg ik hier het idee dat ik er graag met de knex er omheen wilde kunnen gaan bouwen. Wat nu dus niet mogelijk was. Omdat het de exacte afmetingen heeft van de knex vorm die ik gemaakt had.



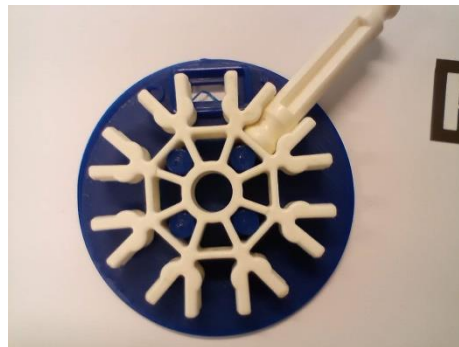
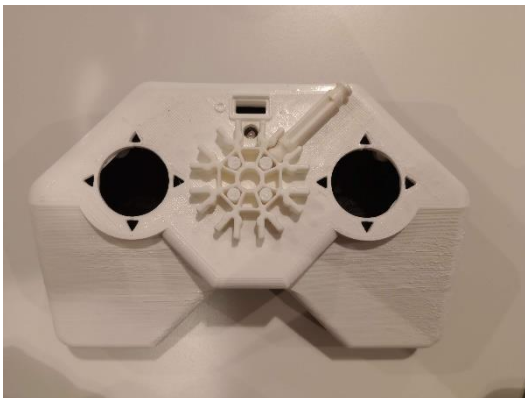
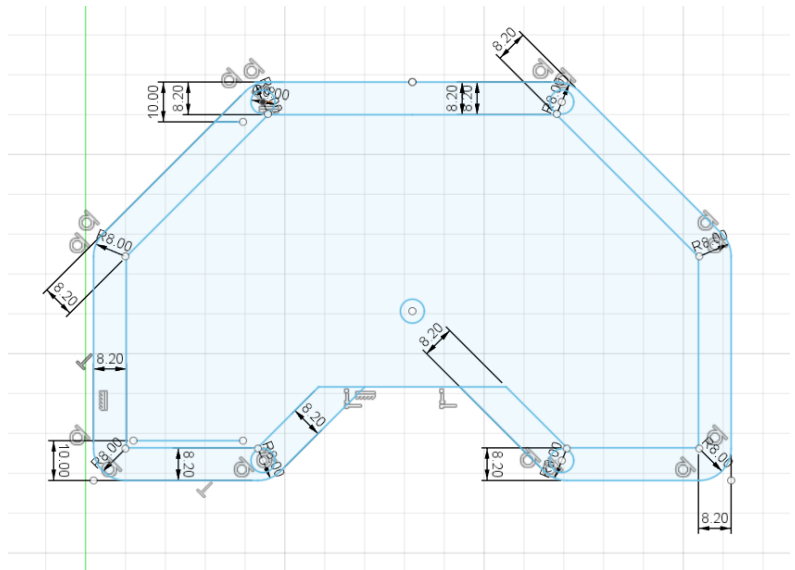
Om die reden heb ik het ontwerp dus een slag kleiner gemaakt. Waardoor ik er als het goed is omheen zou kunnen gaan bouwen.

Voor deze 2<sup>e</sup> versie heb ik ook de pinnen toegevoegd die je op verschillende K'nex onderdelen ziet. Zoals de motor. Deze wilde ik eerst graag testen op een klein stuk omdat dit vrij nauw komt en het perfect moet passen.

Ook had ik gekozen om de pijltjes te vervangen door simpele driehoekjes. Waar ik dan nog een los hoekje in kon doen om hem op te vullen. Met een kleur. Deze driehoekjes zie je ook op de K'nex motoren en is dus al het design van K'nex om richting mee aan te geven. Simpel dus.

Ook het ontwerp voor de aan en uit knop heb ik gekopieerd van de moderne K'nex motor. Om zo dicht bij de vormgeving van K'nex te blijven. Ook dit had ik op het test stukje geprint.

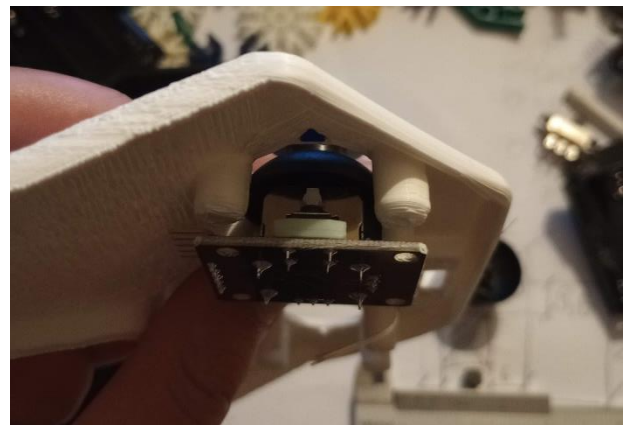
Naast de knoppen wilde ik ook graag nog feedback vanuit de controller. Hoe weet je dat die aan staat? Hoe weet je dat die het doet? Hiervoor heb ik een RGB led toegevoegd om op deze manier zo simpel en minimalistisch mogelijk de feedback te geven. Door een rgb-led te kiezen hield ik ook nog genoeg opties over om veel verschillende signalen af te geven naar de gebruiker.



Boven helft en onder helft. Net als bijna elke controller die op de markt is. Heb ik hem ontworpen met een helft boven en een helft onder. Deze paste na het printen niet heel goed op elkaar. Dit lag uiteindelijk aan de kwaliteit van de printer.

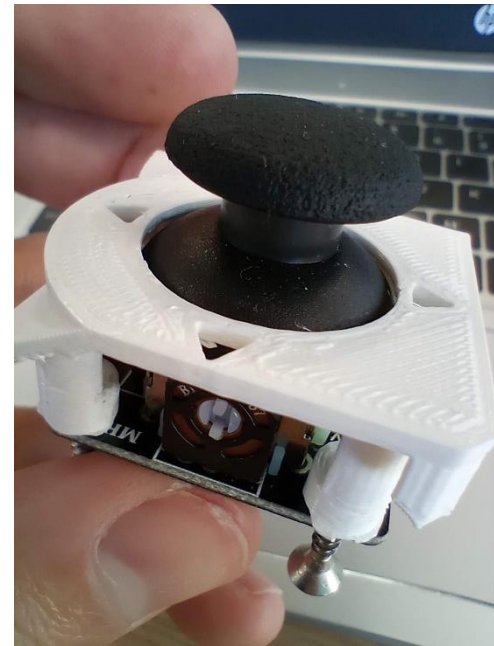
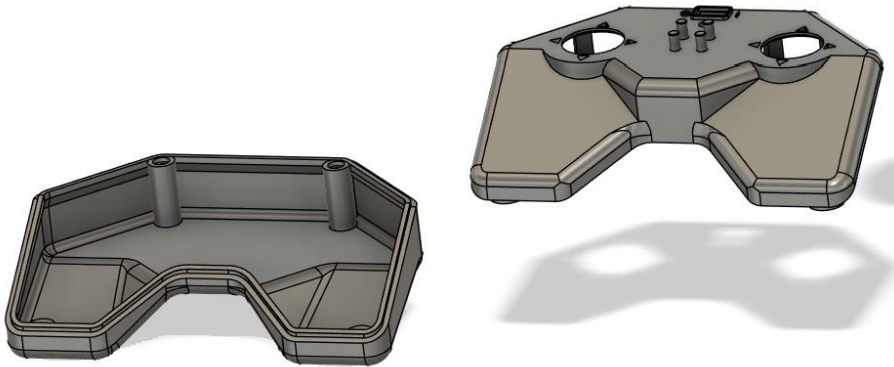
Helaas paste de joystick niet in 1 keer. Ik dacht hiervoor juist gemeten te hebben maar dat was erg lastig. Ook kon ik geen duidelijke afmetingen vinden van het volledige product. Ik heb de leverancier hier ook nog over gecontact. Die konden mij ook niet meer geven als de buitenste maten.

Hiervoor heb ik wel een ontwerp gevonden van iemand die ik uiteindelijk overgetrokken heb.

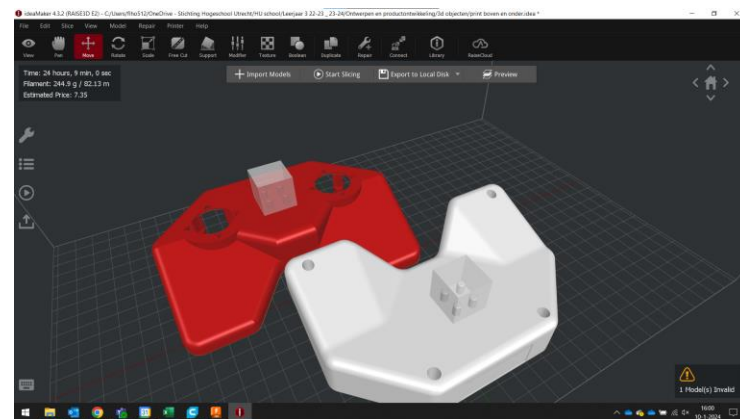


Waarna het uiteindelijk perfect was. Hiervoor heb ik ook eerst een testprint gedaan omdat het anders ook te veel tijd zou kosten.

Uiteindelijk heb ik de aanpassing hiervoor gedaan en nog busjes toegevoegd om de 2 helfden op elkaar te kunnen schroeven. Waarna ik de laatste print heb uitgeprint in het grijs. Ik zag bij de vorige print dat deze al snel vies werd omdat die helemaal wit is.



Het printen van deze versie de boven en onderkant. Duurde in totaal 24 uur.





Uiteindelijk zou het product gespuitsgiet moeten worden. Omdat dit proces veel sneller en preciezer is als het 3d-printen. Voor het ontwerpen en testen van maten kwam het 3d printen wel goed uit. Omdat het maken van een mal veel duurder is en dit nog langer duurt. Als deze mal er uiteindelijk is dan zou dit wel veel sneller kunnen gaan. Misschien dat ik dit in de toekomst nog een keer laten doen.

### 5.1. Arduino code

De code heb ik samen met ChatGPT geschreven. Dit hielp mij om vrij complexe ideeën gemakkelijk toe te voegen.

#### 5.1.1. Controller

```
#include <SPI.h>
#include <nRF24L01.h>
#include <RF24.h>

RF24 radio(9, 10); // CE, CSN pins

int joystick1XPin = A0;
int joystick1YPin = A1;
int joystick2XPin = A2;
int joystick2YPin = A3;

int redPin = 3;
int greenPin = 4;
int bluePin = 5;

unsigned long lastTransmissionTime = 0;
unsigned long lastGreenLedTime = 0;
unsigned long lastRedLedBlinkTime = 0;
const unsigned long transmissionInterval = 20000; // 20 seconden
const unsigned long blueLedDuration = 1000; // 1 seconde
const unsigned long greenLedTimeout = 20000; // 20 seconden
const unsigned long greenLedDuration = 1000; // 1 seconde
const unsigned long redLedBlinkInterval = 20000; // 20 seconden
const unsigned long redLedBlinkDuration = 1000; // Duur van het knipperen van
het rode LED-je

void setup() {
  Serial.begin(9600);
  radio.begin();
  radio.openWritingPipe(0xF0F0F0F0E1LL); // Pipe 0, Adres 0xF0F0F0F0E1LL

  pinMode(redPin, OUTPUT);
  pinMode(greenPin, OUTPUT);
  pinMode(bluePin, OUTPUT);

  // Rode LED
  digitalWrite(redPin, HIGH);
  digitalWrite(greenPin, LOW);
```

```

digitalWrite(bluePin, LOW);
delay(1000); // Wacht 1 seconde

// Groene LED
digitalWrite(redPin, LOW);
digitalWrite(greenPin, HIGH);
digitalWrite(bluePin, LOW);
delay(1000); // Wacht 1 seconde

// Blauwe LED
digitalWrite(redPin, LOW);
digitalWrite(greenPin, LOW);
digitalWrite(bluePin, HIGH);
delay(1000); // Wacht 1 seconde
digitalWrite(bluePin, LOW);
}

void loop() {
    // Lees de waarden van de joysticks
    int joystick1XValue = analogRead(joystick1XPin);
    int joystick1YValue = analogRead(joystick1YPin);
    int joystick2XValue = analogRead(joystick2XPin);
    int joystick2YValue = analogRead(joystick2YPin);

    // Creëer een char-array om de gegevens te verzenden
    char data[8];
    memcpy(data, &joystick1XValue, sizeof(joystick1XValue));
    memcpy(data + sizeof(joystick1XValue), &joystick1YValue,
sizeof(joystick1YValue));
    memcpy(data + 2 * sizeof(joystick1XValue), &joystick2XValue,
sizeof(joystick2XValue));
    memcpy(data + 3 * sizeof(joystick1XValue), &joystick2YValue,
sizeof(joystick2YValue));

    // Verstuur de gegevens via de radio
    radio.write(&data, sizeof(data));

    // Controleer of de joystickgegevens buiten het bereik [400, 600] vallen
    if ((joystick1XValue < 400 || joystick1XValue > 600) ||
        (joystick1YValue < 400 || joystick1YValue > 600) ||
        (joystick2XValue < 400 || joystick2XValue > 600) ||
        (joystick2YValue < 400 || joystick2YValue > 600)) {
        // Zet het groene LED-je aan als de joystickgegevens buiten het bereik
vallen
        digitalWrite(greenPin, HIGH);
        lastGreenLedTime = millis(); // Update de tijd van de laatste keer dat het
groene LED-je is aangezet
    } else {

```

```

    // Zet het groene LED-je uit als de joystickgegevens binnen het bereik
    vallen
    digitalWrite(greenPin, LOW);
}

unsigned long currentTime = millis();
// Schakel het rode LED-je 1 seconde aan elke 20 seconden als het groene
LED-je niet heeft gebrand
if ((currentTime - lastGreenLedTime) >= greenLedTimeout) {
    // Het groene LED-je heeft 20 seconden gebrand, dus knipper het rode LED-
je
    unsigned long elapsedTime = currentTime - lastRedLedBlinkTime;
    if (elapsedTime <= redLedBlinkDuration) {
        digitalWrite(redPin, HIGH);
    } else {
        digitalWrite(redPin, LOW);
    }

    if (elapsedTime >= (redLedBlinkDuration + redLedBlinkInterval)) {
        lastRedLedBlinkTime = currentTime;
    }
} else {
    digitalWrite(redPin, LOW);
}

// Update de tijd van de laatste transmissie
lastTransmissionTime = currentTime;
// Wacht voor een korte periode voordat je opnieuw leest en verzendt
delay(100);
}

```

### 5.1.2. Ontvanger

```

#include <SPI.h>
#include <nRF24L01.h>
#include <RF24.h>

RF24 radio(9, 10); // CE, CSN pins
int joystick1XValue, joystick1YValue, joystick2XValue, joystick2YValue;

const int motor1PWM = 3; // PWM-snelheidsregeling voor Motor 1
const int motor1Dir1 = 4; // Richting 1 voor Motor 1
const int motor1Dir2 = 5; // Richting 2 voor Motor 1

const int motor2PWM = 6; // PWM-snelheidsregeling voor Motor 2
const int motor2Dir1 = 7; // Richting 1 voor Motor 2
const int motor2Dir2 = 8; // Richting 2 voor Motor 2

const int motor3PWM = 11; // PWM-snelheidsregeling voor Motor 3

```

```

const int motor3Dir1 = 12; // Richting 1 voor Motor 3
const int motor3Dir2 = 13; // Richting 2 voor Motor 3

const int motor4PWM = A0; // PWM-snelheidsregeling voor Motor 4
const int motor4Dir1 = A1; // Richting 1 voor Motor 4
const int motor4Dir2 = A2; // Richting 2 voor Motor 4

void setup() {
    Serial.begin(9600);
    radio.begin();
    radio.openReadingPipe(1, 0xF0F0F0F0E1LL); // Pipe 1, Adres 0xF0F0F0F0E1LL
    radio.startListening();

    pinMode(motor1PWM, OUTPUT);
    pinMode(motor1Dir1, OUTPUT);
    pinMode(motor1Dir2, OUTPUT);
    pinMode(motor2PWM, OUTPUT);
    pinMode(motor2Dir1, OUTPUT);
    pinMode(motor2Dir2, OUTPUT);
    pinMode(motor3PWM, OUTPUT);
    pinMode(motor3Dir1, OUTPUT);
    pinMode(motor3Dir2, OUTPUT);
    pinMode(motor4PWM, OUTPUT);
    pinMode(motor4Dir1, OUTPUT);
    pinMode(motor4Dir2, OUTPUT);
}

void loop() {
    if (radio.available()) {
        // Ontvang de gegevens van de verzender
        char data[8];
        radio.read(&data, sizeof(data));

        // Haal de waarden van de joysticks op uit de ontvangen gegevens
        memcpy(&joystick1XValue, data, sizeof(joystick1XValue));
        memcpy(&joystick1YValue, data + sizeof(joystick1XValue),
            sizeof(joystick1YValue));
        memcpy(&joystick2XValue, data + 2 * sizeof(joystick1XValue),
            sizeof(joystick2XValue));
        memcpy(&joystick2YValue, data + 3 * sizeof(joystick1XValue),
            sizeof(joystick2YValue));

        Serial.print(data);

        // Verwerk de ontvangen waarden (je kunt hier bijvoorbeeld de waarden
        afdrukken)
        // Serial.print("test");
    }
}

```

```

// Stel de snelheid van de motoren in op basis van de joystickwaarden
int motor1Speed = map(joystick1XValue, 0, 1023, -255, 255); // Joystick 1X
voor Motor 1
int motor2Speed = map(joystick1YValue, 0, 1023, -255, 255); // Joystick 1Y
voor Motor 2
int motor3Speed = map(joystick2XValue, 0, 1023, -255, 255); // Joystick 2X
voor Motor 3
int motor4Speed = map(joystick2YValue, 0, 1023, -255, 255); // Joystick 2Y
voor Motor 4

// Beperk de snelheid binnen het bereik van -255 tot 255
motor1Speed = constrain(motor1Speed, -255, 255);
motor2Speed = constrain(motor2Speed, -255, 255);
motor3Speed = constrain(motor3Speed, -255, 255);
motor4Speed = constrain(motor4Speed, -255, 255);

//Motor 1
if (motor1Speed > 0) {
    analogWrite(motor1PWM, motor1Speed);
    digitalWrite(motor1Dir1, HIGH);
    digitalWrite(motor1Dir2, LOW);
} else {
    analogWrite(motor1PWM, abs(motor1Speed));
    digitalWrite(motor1Dir1, LOW);
    digitalWrite(motor1Dir2, HIGH);
}
//Motor 2
if (motor2Speed > 0) {
    analogWrite(motor2PWM, motor2Speed);
    digitalWrite(motor2Dir1, HIGH);
    digitalWrite(motor2Dir2, LOW);
} else {
    analogWrite(motor2PWM, abs(motor2Speed));
    digitalWrite(motor2Dir1, LOW);
    digitalWrite(motor2Dir2, HIGH);
}
//Motor 3
if (motor3Speed > 0) {
    analogWrite(motor3PWM, motor3Speed);
    digitalWrite(motor3Dir1, HIGH);
    digitalWrite(motor3Dir2, LOW);
} else {
    analogWrite(motor3PWM, abs(motor3Speed));
    digitalWrite(motor3Dir1, LOW);
}

```



```

        digitalWrite(motor3Dir2, HIGH);
    }
    //Motor 4
    if (motor2Speed > 0) {
        analogWrite(motor4PWM, motor4Speed);
        digitalWrite(motor4Dir1, HIGH);
        digitalWrite(motor4Dir2, LOW);
    } else {
        analogWrite(motor4PWM, abs(motor4Speed));
        digitalWrite(motor4Dir1, LOW);
        digitalWrite(motor4Dir2, HIGH);
    }
}
else {
    delay(10);
}
}
}

```

## 6. Productievoorbereiding

De Arduino's heb ik uiteindelijk bij mijn werk mee kunnen nemen. De Arduino pro mini's deze worden ook gebruikt om het project drones te maken. Het is een kleine Arduino met voldoende in en outputs. Deze heeft geen usb aansluiting wat hem dan ook weer een stuk voordeliger maakt en compacter.

Veder heb ik alle onderdelen besteld bij Tinytronics. Deze leverancier levert snel en is betrouwbaar. Prima prijzen.

Ik heb er meerdere batterijhouders bij gedaan omdat ik nog niet zeker wist wat er ging werken en passen.

# TINYTRONICS

TinyTronics B.V.  
Ruysdaelbaan 3E  
5642 JJ Eindhoven  
the Netherlands

Website: <https://www.tinytronics.nl/shop>

## FACTUUR

**Ordernummer:** 428722  
**Besteldatum:** 05-01-2024  
**Factuurnummer:** INV-2024-1070  
**Factuurdatum:** 05-01-2024  
**Betaalwijze:** IDEAL  
**Verzendwijze:** PostNL Ophaalpunt NL

Factuuradres	Verzendadres
Florent Hofman [REDACTED]	PostNL Pakketpunt Pakketautomaat S.M. Hugo van Gijnweg 2PBA 3314HX Dordrecht Nederland

Product	SKU	Aantal	Prijs	Prijs ex. BTW	Totaal ex. BTW
PS2 Joystick	000156	2	€2,50	€2,07	€4,13
Inbouw Schakelaar ON-OFF-ON - SS23F19	002063	3	€0,60	€0,50	€1,49
3x AA Batterijhouder met Losse Draden	000516	1	€1,00	€0,83	€0,83
4x AAA Batterijhouder met Losse Draden	001457	1	€1,40	€1,16	€1,16
4x AA Batterijhouder met Losse Draden	000854	1	€1,25	€1,03	€1,03
3x AAA Batterijhouder met Losse Draden	003342	1	€1,00	€0,83	€0,83
L298N DC-Motoraansturing	000117	2	€5,00	€4,13	€8,26
8x AA Batterijhouder met Losse Draden	000688	1	€2,50	€2,07	€2,07
6x AA Batterijhouder met Losse Draden	000687	1	€1,50	€1,24	€1,24
SynFlex W 210 Gelakt Koperdraad - 1.0mm - 250g	003765	1	€12,50	€10,33	€10,33
<b>Subtotaal:</b>					€31,36 Incl. BTW: €37,95
<b>PostNL Ophaalpunt NL:</b>					€4,92 Incl. BTW: €5,95
<b>NL BTW/VAT (21%):</b>					€7,62
<b>Totaal:</b>					€43,90

Product	SKU	Aantal	Prijs	Prijs ex. BTW	Totaal ex. BTW
Inbouw Schakelaar ON-OFF - SS22G08	002062	4	€0,60	€0,50	€1,98
<b>Subtotaal:</b>					€1,98 Incl. BTW: €2,40
<b>Brievenbuspost Standaard Nederland:</b>					€2,07 Incl. BTW: €2,50
<b>NL BTW/VAT (21%):</b>					€0,85
<b>Totaal:</b>					€4,90

## 7. Evaluatie en testfase

### Het testen van het product

Ik heb tijdens het programmeren zelf al veel getest.

Als je de controller en de ontvanger aanzet dan beginnen de motoren al een beetje te brommen. Dit moet nog opgelost worden in de code. Dit komt waarschijnlijk omdat de ontvanger al wel wat ontvangt en dat ook gelijk al doorstuurt. Terwijl de controller nog helemaal niet bedient wordt. Maar dit ligt net boven of onder de waarde waar die niks stuurt. Waardoor hij dus wel wat stuurt maar dit niet genoeg is voor de motoren om te gaan draaien. Er moet dus een groter dood punt in geprogrammeerd worden. Deze zit er nu eigenlijk nog niet in.

Het testen op de gebruikersgroep moet nog uitgebreider gebeuren.

De controller ligt wel al goed in de hand voor de gebruikers vanaf 18 jaar. Deze feedback kreeg ik terug na de laatste pitch. Ook kreeg ik terug dat het als kwaliteit aanvoelde. Dit door het gewicht. Hij zit volledig vol wat hem dit gewicht dus ook geeft.

Hoe kinderen vanaf 5 jaar er mee om gaan ga ik nog testen als de ontvanger af is.

## 8. Definitiefase

Het volledig in productie brengen van het product gaat waarschijnlijk niet gebeuren. Wel ga ik als de ontvanger af is het product delen op Reddit. Waarbij ik een link met de projectinformatie wil toevoegen. Op welke website ik dit ga zetten moet ik nog onderzoeken. Ik wil het zo klaar zetten dat iedereen hem eventueel ook thuis na zou kunnen maken.

Het idee verkopen aan K'nex wordt hem waarschijnlijk niet. Omdat een controller zelf te simpel is. Ook hebben ze geen duidelijke contact gegevens online staan.

## 9. Didactische koppeling

### 9.1. Projecten

Voornamelijk heb ik bij het maken van de K'nex controller, veel geleerd over het ontwerpproces. Dit is eigenlijk iets waar ik op mijn eigen school ook veel mee bezig ben met de studenten. Maar nu stond ik er zelf in. In plaats van de zijlijn.

Loopt van stap naar stap. Je doet iets omdat het nodig is niet omdat het moet. Wat bij de studenten soms wel eens anders is. Hier zouden wij in ons onderwijs nog het een en ander kunnen verbeteren. We moeten ze veel verschillende vormen aanbieden. En ze soms ook vrij houden in hun onderzoek. Waardoor er misschien wel meer uit komt. Ze zouden meer gefocust moeten zijn op het uiteindelijke product in plaats van de onderdelen die er tussen komen die voor de docent gemaakt moeten worden.

Vaak wordt er ook geklaagd over de eerste paar fases binnen het project. Dat dit te veel papier werk is. Maar dit zou naar mijn inzicht en zeker ook na deze ervaring een stuk beter en leuker kunnen.

Hierin zou een goed tussenweg gevonden moeten worden. Waarbij we ze opleiden om zelfstandig hier doorheen te kunnen lopen en het effect van de stappen ook echt voelen en zien. In plaats van dat ze het voor de docent doen.

### 9.2. Fushion360

Ook in het programma fushion360 ben ik nu vaardig. Waardoor ik de studenten nu ook de weg kan wijzen.