Rapport EE3

# Inleiding – samenvatting

Na drie introductielessen over LabVIEW en vier seminaries over de werking van Computer-Based Control (CBC) is onze groep zelf aan de slag gegaan om een geautomatiseerde robot in elkaar te steken en te programmeren. We kregen een zevental weken, die ingedeeld waren in drie sprints, om ons project tot een succesvol einde te brengen. Aan de hand van door ons vooraf bepaalde taken konden we (bijna) altijd de opgelegde deadlines van deze sprints halen.

De bedoeling van dit vak was om zelf een CBC te ontwerpen. Na enkele brainstormsessies hadden we besloten een zelfrijdende robot te maken, uitgerust met een grijparm en sensoren. De functie van deze robot is een aantal blokjes in een bepaalde kleur, die de gebruiker als input ingeeft, te verzamelen. Men kan dus zelf bepalen hoeveel blokjes van een bepaalde kleur (rood, groen of blauw) het robotje moet gaan halen. Zo kan men bijvoorbeeld vragen aan de robot om drie groene en twee rode blokjes te gaan halen waarna de robot in activeert en begint te rijden. Door middel van een afstandssensor en een kleurensensor kan de robot detecteren wanneer het aan een blokje is aangekomen en welke kleur dit blokje heeft. Indien de kleur van het blokje overeenkomt met de gewenste kleur (en rekening houdend met het aantal), treedt de grijper automatisch in werking om het blokje vast te nemen en te deponeren in een bakje. Hierna rijdt de robot weer verder naar het volgende blokje of, indien het aantal gewenste blokjes al verzameld is, keert het terug naar de startpositie.

In dit verslag vindt men alle informatie over ons zelfgemaakte robotje terug. Het verslag is opgedeeld in twee grote segmenten en een conclusie. Als eerste komt het engineering gedeelte aan bod. Hier bespreken we de productvereisten, waarna dieper ingaan wordt op het proces van de conceptgeneratie. Na het voorstellen van de verschillende begin concepten en het uiteindelijke gekozen concept, behandelen we de specificaties over de mechanische werking van ons finale prototype. Ook de bespreking van de gebruikte actuatoren, sensoren en het programma komen hier aan bod.

Daarna voeren we in het enterprising gedeelte een kostprijsberekening uit en analyseren we deze kosten op efficiëntie, om er zo achter te komen hoe we het project beter hadden kunnen aanpakken. Ook onderzoeken we onze motivatiecurves, zowel individueel als die voor het gehele team en trekken we hier conclusies uit.

We eindigen het verslag met een besluit, waar we het eindresultaat van het project bespreken. We komen tot de conclusie dat het een werkende automatische robot, die voldoet aan de eisen en gepast kan reageren op de gewenste input, is. Op vlak van efficiëntie en teamwerk kunnen we echter besluiten dat er nog ruimte was voor verbetering. Zowel tijd als kosten kunnen bespaard zijn geweest, moesten we het project vanaf het begin anders hebben aangepakt. Het project heeft ons op vlak van efficiëntie en teamwerk dus wel veel bijgeleerd.

# Engineering gedeelte

## Productvereisten

Opruimen en sorteren is voor vele mensen een heel gedoe tegenwoordig. Daarbovenop maakt de toenemende luiheid van onze samenleving het er niet veel beter op. Daarom trachten wij een robot te ontwerpen die als precedent voor toekomstige opruimrobots kan dienen. Het concept is als volgt; een robotje fabriceren dat aan de hand van twee verschillende inputs (kleur en aantal) zelfstandig, al rijdend binnen een afgebakend gebied, blokjes ophaalt en sorteert.

### Lijst met vereisten

|  |  |  |  |
| --- | --- | --- | --- |
|  | Omschrijving | Eisen | Wensen |
| Gebruik van Fischertechnik | - | x |  |
| Gebruik van LabVIEW | Eindigetoestandsautomaat | x |  |
| Feedback | Gebruik van (verschillende) sensors voor feedback | x |  |
| Integratie van motors | Maximum van vier motors | x |  |
| Veiligheid | Geen 220V, ≤ 48V | x |  |
| Compact design | Gemakkelijk op te bergen en te transporteren |  | x |
| Origineel concept | Geen *pick and place* robot |  | x |
| Gemakkelijk te vervaardigen | Standaard materialen en sensoren |  | x |
| Budgetvriendelijk | Minimale geldsom aan het project besteden |  | x |
| Correcte uitvoering | De robot brengt zijn voorbedachte taak tot een goed einde | x |  |

### Functionele afbraak

* Identificatie; kleuren herkennen en deze vergelijken met de input die is gegeven
* Verplaatsing; zelfstandig naar het doel en terug rijden
* Grijpen; de beweging uitvoeren om een blokje vast te pakken
* Dragen; het blokje vasthouden
* Plaatsen; het blokje in de bak deponeren voor verder transport
* Doelen herkennen; ‘waar blokjes ophalen en retourneren?’
* Tellen; de hoeveelheid blokjes vergelijken met het aantal dat is gegeven als input

## Conceptgeneratie

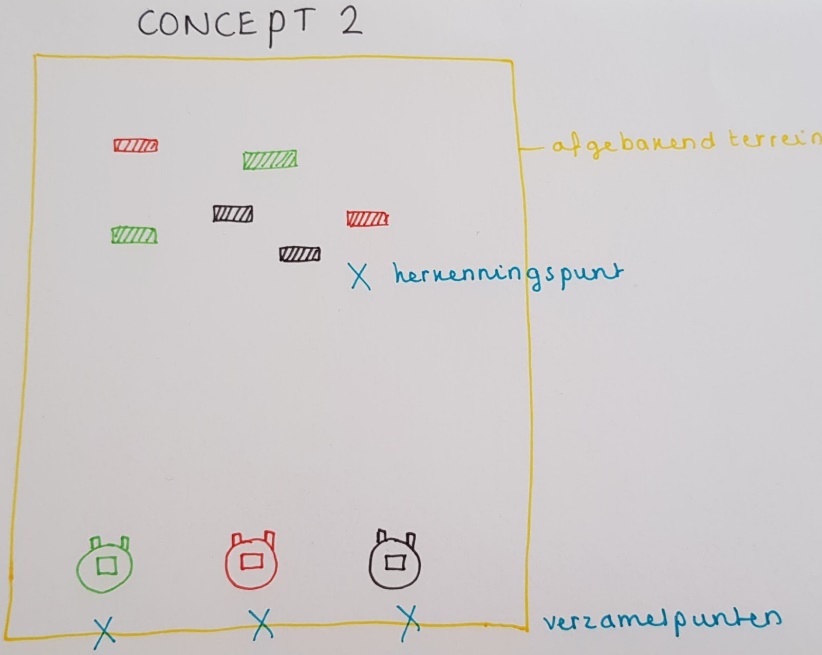
Voor de simpliciteit zal bij de waardeanalyse van elk concept enkel datgene in kaart worden gebracht dat specifiek is voor het concept. Zo zal er geen rekening worden gehouden met bijvoorbeeld de kosten van de blokjes, aangezien elk concept hier gebruik van maakt.

### https://lh5.googleusercontent.com/kXB9BEm5DXD6lkHff48In5B4PEq9qiyABhvIyBHRJUJQrhI_Kz7lT-2qHbuaTomP8JLaA37duwFtJ_dNSU4bev9-olNupmc9zkNZCV-hspKfHzlq9K4IwLQrulUi6rGeDmR5bRfoConcept 1: Eén wagentje voor alle kleuren

Het eerste concept maakt gebruik van één wagen/robot die met behulp van sensoren in een afgebakend gebied, zelfstandig zijn weg vindt naar een bepaald herkenningspunt. Eenmaal dat hij daar aankomt grijpt hij afzonderlijk de gewenste blokjes (afhankelijk van de gegeven input) en deponeert hij die blokjes in de correcte bak (deze bakken zijn geïntegreerd in het design). Als de robot ten slotte alle gewenste blokjes heeft verzameld rijdt hij terug naar het verzamelpunt/startpunt met de opgegrepen blokjes.

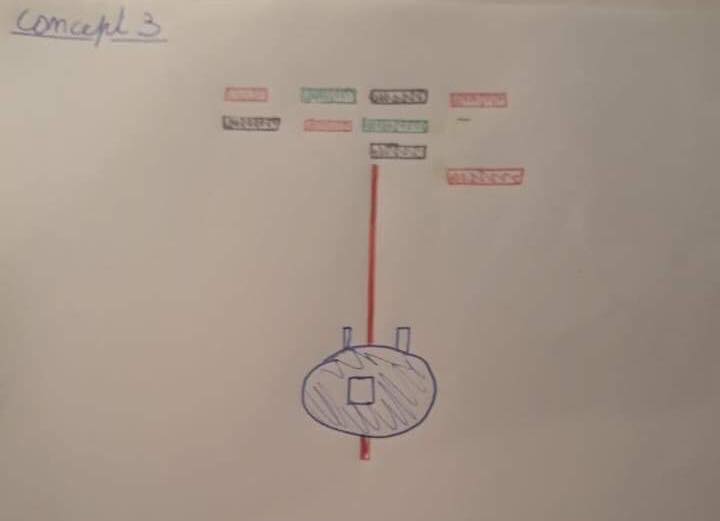
Bij concept 1 zou het robotje aan de hand van een webcam de blokjes opsporen. De prijs van zo’n webcam bedraagt ongeveer 25 euro.

### Concept 2: Eén wagentje per kleur

****In het tweede concept maken we gebruik van drie verschillende wagentjes. Elk wagentje correspondeert zo met een specifieke kleur. Ze rijden echter nog steeds autonoom binnen een afgebakend gebied. Net zoals het eerste concept, wordt dit concept gerealiseerd aan de hand van sensoren, een herkenningspunt en een verzamelpunt. In tegenstelling tot het eerste concept, staat elk robotje in voor het tellen van een specifieke kleur. Zo zoekt bijvoorbeeld een rood wagentje de rode blokken en een gele de gele blokken. Deze blokken plaatsen ze dan ook in een bak dat deel uitmaakt van de robot zelf. Het aantal blokken en de hoeveelheid per kleur hangt natuurlijk nog steeds af van de gegeven input.

Bij dit concept worden er dus drie robotjes gebruikt wat maakt dat de kosten al snel zullen oplopen. De waarde van een robotje wordt geschat rond 330 euro, wat maakt dat men al meteen met 660 euro extra kosten zit. Daarbovenop moet elk wagentje nog eens uitgerust zijn met een webcam, wat de totale extra prijs (naast het basisconcept) laat oplopen tot 735 euro.

### Concept 3: Eén wagentje dat een vast traject volgt en elk blokje apart gaat ophalen

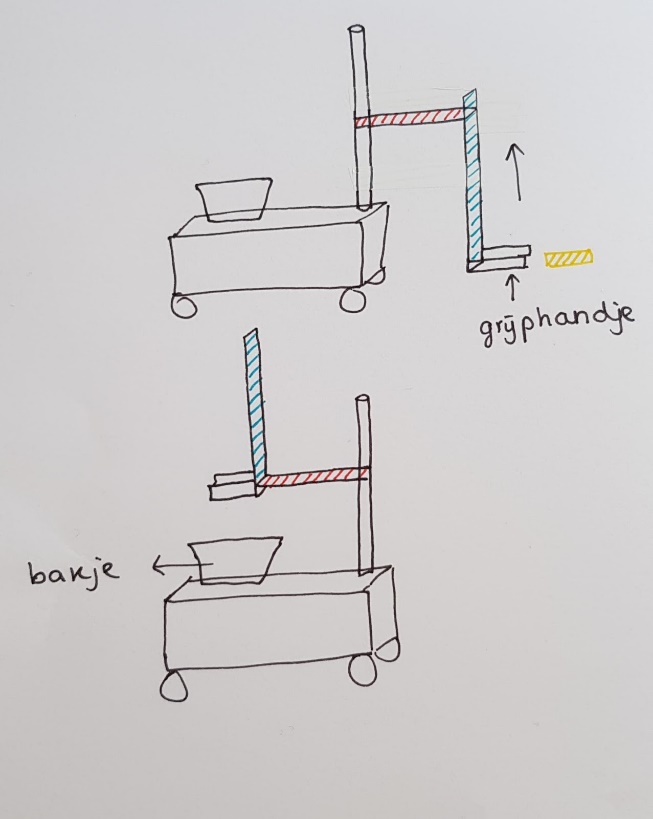
****Bij dit concept volgt een robot een vast traject dat op de grond getekend staat (met behulp van kleursensoren die bijvoorbeeld een zwarte lijn herkennen). De robot volgt deze lijn tot aan de blokken (we maken dus niet gebruik van een herkennings-/verzamelpunt). Eenmaal gearriveerd zal de robot een blokje dat hij nodig heeft nemen en deze stevig vasthouden. Vervolgens zal de robot langs dezelfde afgetekende lijn met het correcte blokje terugkeren en in de juiste bak leggen. Dit proces wordt herhaald totdat alle gevraagde stukken gehaald zijn (afhankelijk van de input).

Bij het laatste concept zou er geen gebruik gemaakt worden van een webcam maar van een lichtsensor. Zo’n LDR (light-depending resistor) kost ongeveer 20 eurocent per stuk.

### Evaluatietabel

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Vorm van evaluatie | De robot… | Concept 1 | Concept 2 | Concept 3 |
| Zelfrijdend | kan zich autonoom verplaatsen | x | x | x |
| Objectherkenning | kan de blokken zelfstandig vinden | x | x |  |
| Kleur herkenning | kan verschillende kleuren herkennen | x |  | x |
| Prijskaartje | kost relatief weinig | x |  | x |

### Gekozen concept

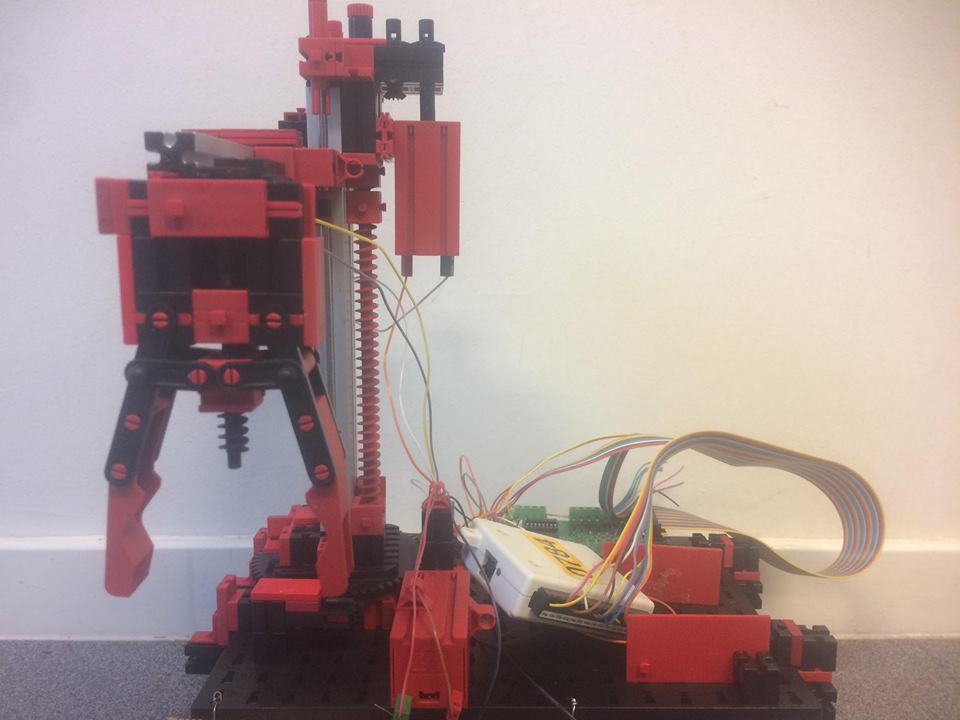
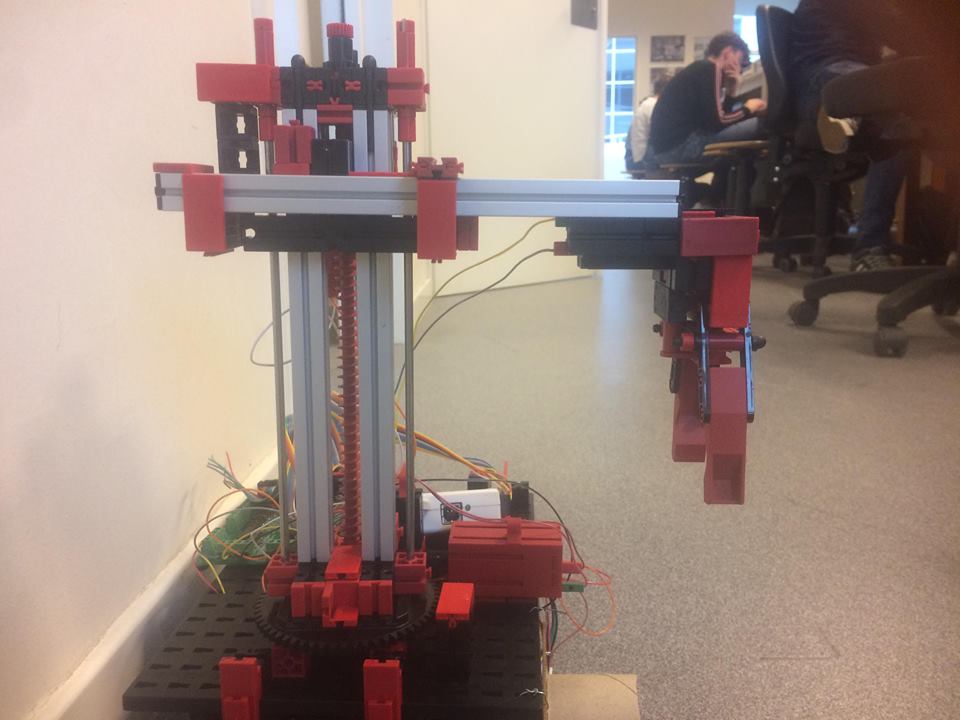
We kiezen voor het **eerste concept** aangezien we ervan overtuigd zijn dat deze de meest efficiënte manier is om de gevraagde blokjes te gaan halen en sorteren. Daarbovenop lijkt het ons ook interessanter om de robot zelfrijdend te maken in plaats van het een lijntje te laten volgen.

In de tekening ziet men een schets ter verduidelijking van de grijparm van de robot. De arm zal dus drie functies bevatten: omhoog en omlaag bewegen, draaien rond een vaste as en het grijphandje open en dicht laten gaan. Zoals op de tekening valt te zien, bevinden de transportbakjes voor de blokjes zich op de achterkant van de robot.

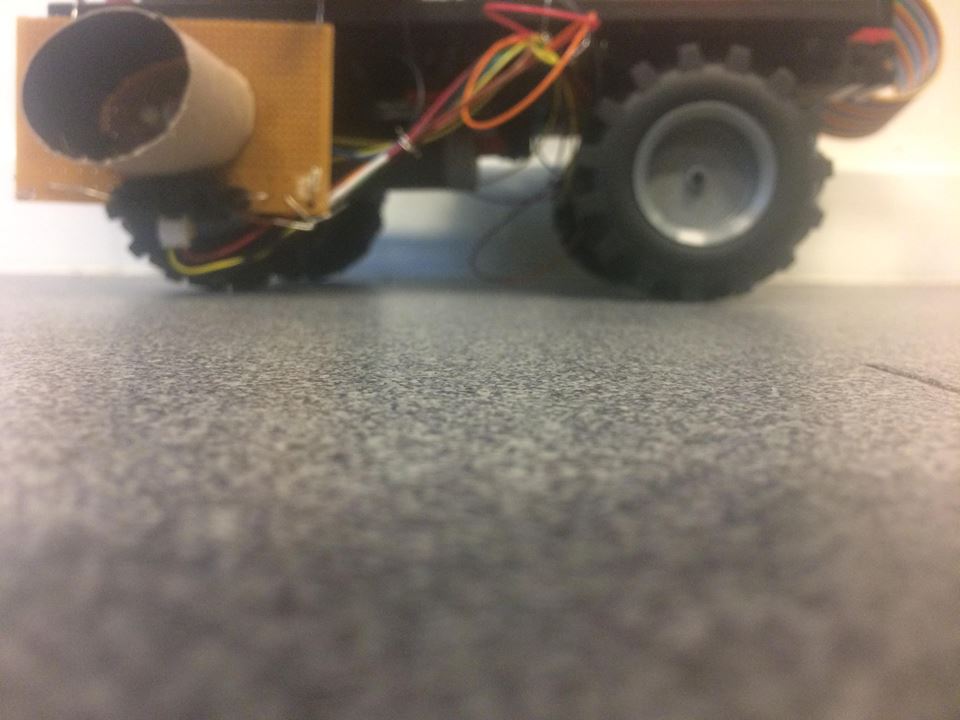
n.v.d.r.: Uiteindelijk hebben we nog enkele zaken van het gekozen concept aangepast. De voornaamste verandering is dat het robotje niet meer sorteert binnenin een afgebakend gebied. We gaven de voorkeur aan het optimaliseren van de basisfuncties waardoor we geen tijd meer hadden om de extra functies te implementeren. Bovendien vereiste het draaien van de wielen een extra motor. Verder werkt het robotje met sensoren in plaats van herkenningspunten en hebben we ervoor gekozen om het grijphandje verticaal in plaats van horizontaal (zoals in de tekening) te plaatsen.

## Gedetailleerd ontwerp en prototype

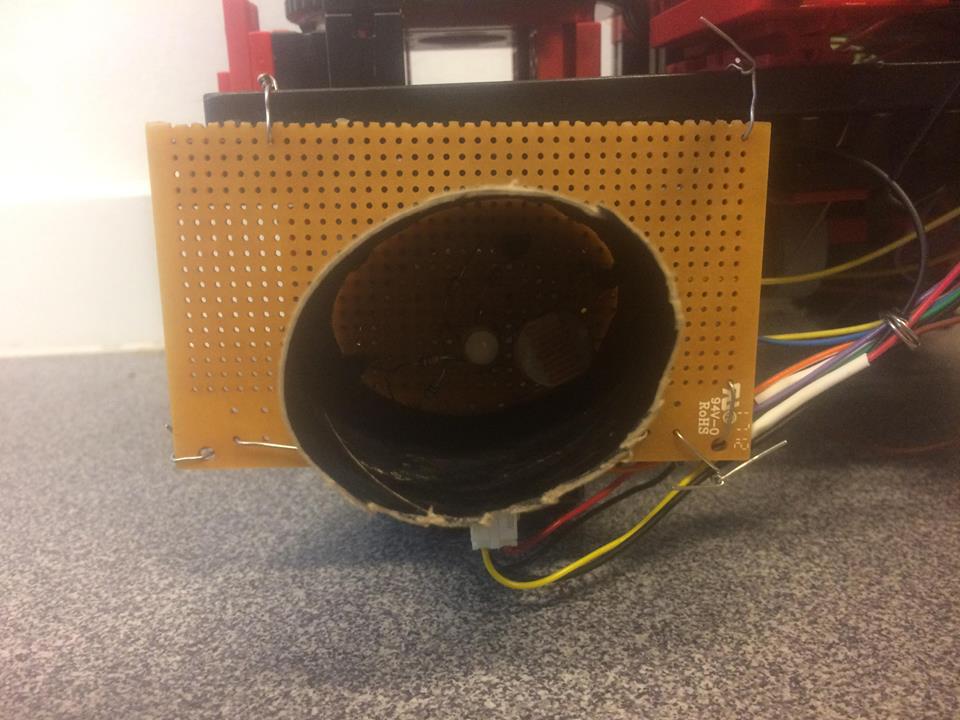
### Mechanische constructie

Onze robot is opgebouwd uit verschillende onderdelen. Het eerste segment is de grijparm. Deze kan omhoog en omlaag bewegen, draaien en een voorwerp vastgrijpen. Om dit te kunnen realiseren maken we gebruik van drie motoren met elks drie toestanden (00/11[[1]](#footnote-1), 01 en 10). Motor 1 laat de grijper omhoog en omlaag bewegen. Dit is noodzakelijk om het hoogteverschil tussen de grond en het bakje te overbruggen. Motor 2 wordt gebruikt om een blokje te kunnen vastgrijpen en loslaten. De laatste motor van de grijper wordt gebruikt om de grijparm te laten roteren. De grijper kan zowel tegen- als kloksgewijs draaien. De bovenconstructie is gemonteerd op een aparte wielconstructie die ervoor zorgt dat de robot kan rijden.

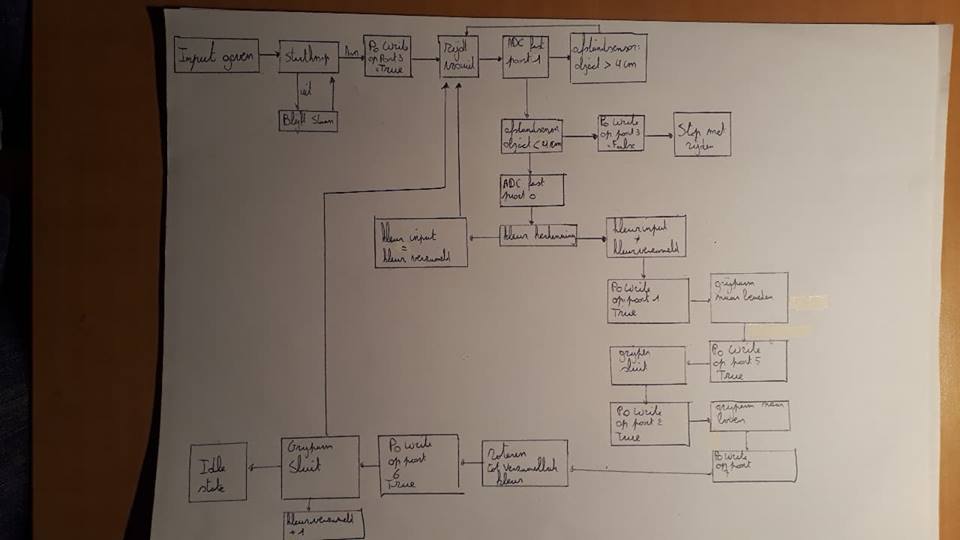
 

Het tweede onderdeel van de robot is het onderstel. Het onderstel bestaat uit één motor en vier wielen. De motor en wielen zijn verbonden met de nodige stangen en tandwielen. De motor heeft drie standen: 01 om de robot vooruit te laten bewegen, 10 om de robot achteruit te laten bewegen en 00/11 om de robot te laten stilstaan. Wanneer het robotje gevraagd wordt om een taak uit te voeren, zal de robot vooruitrijden. Wanneer het de gevraagd taak heeft volbracht zal het robotje achteruit bewegen totdat het zich terug bij het startpunt bevindt. Het onderstel van de robot speelt een cruciale rol in het ontwerp aangezien de robot zonder een stabiel onderstel niet kan bewegen.



Het derde en laatste grote onderdeel van de robot zijn de sensoren. De robot is uitgerust met een afstandssensor die een signaal (een spanningsverandering) verstuurt indien er zich een object voor het robotje bevindt. Met behulp van een kleurensensor scant de robot de kleur van het geplaatste blokje en berekent het aan de hand van de meegegeven inputs of die het blokje moet oppakken of niet. Indien niet, dan zal het robotje gewoon verder rijden tot het volgende blokje.

### Blok schema

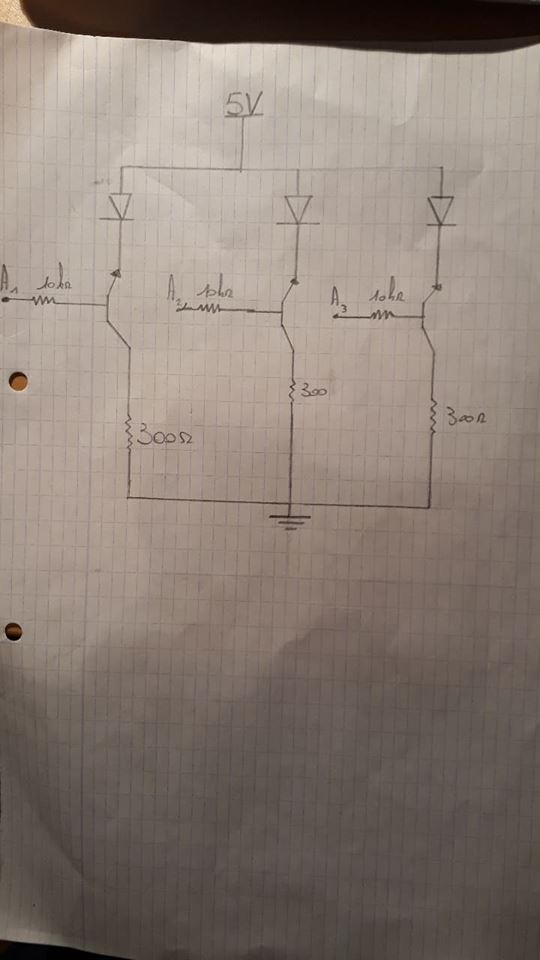


### Sensoren

Voor de kleurherkenning van de blokjes hebben we een LDR gebruikt. Om dit te kunnen realiseren hebben we een RGB-led verbonden en een programma geschreven om de led afwisselend rood, groen en blauw uit te laten stralen. Deze RGB-led hebben we vastgemaakt aan drie transistoren die met hun base verbonden zijn aan een weerstand van 10 kΩ en aan hun emitter verbonden zijn met een 330 Ω weerstand die geconnecteerd is met de grond. Verder zijn de basen van de transistoren telkens verbonden aan een digitale ingang waardoor het programma in LabVIEW ervoor kan zorgen dat er om de beurt 5V aan de base van elke transistor komt. Hierdoor zal de RGB-led de kleur weergeven van de transistor waarbij er 5V over de base staat. Afhankelijk van het opgevangen licht zal de spanning over de LDR toe- of afnemen. Zo geeft het programma een andere waarde weer afhankelijk van de kleur van het blokje dat zich voor de LED bevindt. Voor een grotere nauwkeurigheid hebben we rond de LDR een donkere koker bevestigd. Het nut van deze “kleursensor” is om de kleur van een blokje te herkennen en zo de gevraagde blokjes te kunnen onderscheiden en oppakken. Zonder deze kleurensensor kan er enkel bepaald worden of er zich iets voor het robotje bevindt.

Als alternatief hadden we ook een webcam kunnen gebruiken, maar we hebben hier niet voor gekozen omdat een webcam duurder is en de robot zijn omgeving niet moet kunnen analyseren (enkel de kleuren).

De afstandssensor werkt op basis van infrarood. De afstandssensor die we gebruiken heeft een reikwijdte van vier tot 30 cm. Dit betekent dat indien er een voorwerp op ongeveer vier centimeter van de sensor staat, de sensor een spanning van rond de 5V zal geven. Indien 30 cm, zal de sensor een spanning van rond de 1V geven.

De sensor stuurt een infrarood signaal uit en afhankelijk van de afstand gaat de reflectiehoek verschillen. Indien een object dichter staat gaat de hoek groter zijn dan indien een object verder weg staat. Afhankelijk van deze hoek wordt er dan een bepaalde spanning gemeten, dat weergeeft of het robotje de aanwezigheid van een object opmerkt. De afstandssensor heeft 3 ingangen: de eerste ingang is verbonden aan de analoge ingang, de tweede is verbonden aan de grond en de laatste ingang is geconnecteerd met de ingang van 5V.

Als alternatief voor de afstand sensor zouden we hier ook een webcam kunnen gebruiken. Het principe is hetzelfde als voor een infraroodsensor; als het robotje een voorwerp ziet zal hij stoppen en analyseren of het het gewenste voorwerp is. We hebben hier echter niet voor gekozen omdat, naast de eerder vermelde redenen, een afstandssensor ook veel compacter is.

### Actuatoren

Een actuator is een component in de robot die verantwoordelijk is voor het bewegen en het controleren van het mechanisme. Het bevat een input signaal en een energiebron.

Dit betekent natuurlijk dat elke motor dat gebruikt wordt in ons systeem een actuator is, aangezien per slot van rekening een motor een deel van het systeem in beweging brengt. We gebruiken vier motoren om de robot met behulp van feedback te laten functioneren. Al de motors die men in het robotje terugvindt zijn ‘rotary actuators,’ of specifieker ‘electric rotary actuators,’ wat betekent dat een voltage aan de input gebruikt wordt als output om een deel van het systeem te laten roteren. Met behulp van tandwielen en andere mechanische componenten kan men deze veroorzaakte draaibeweging gemakkelijke overbrengen op verschillende componenten. Indien men de polen van de motoren omwisselt, zal de draaibeweging ook tegengesteld werken, gegeven dat men niets verandert aan de input.

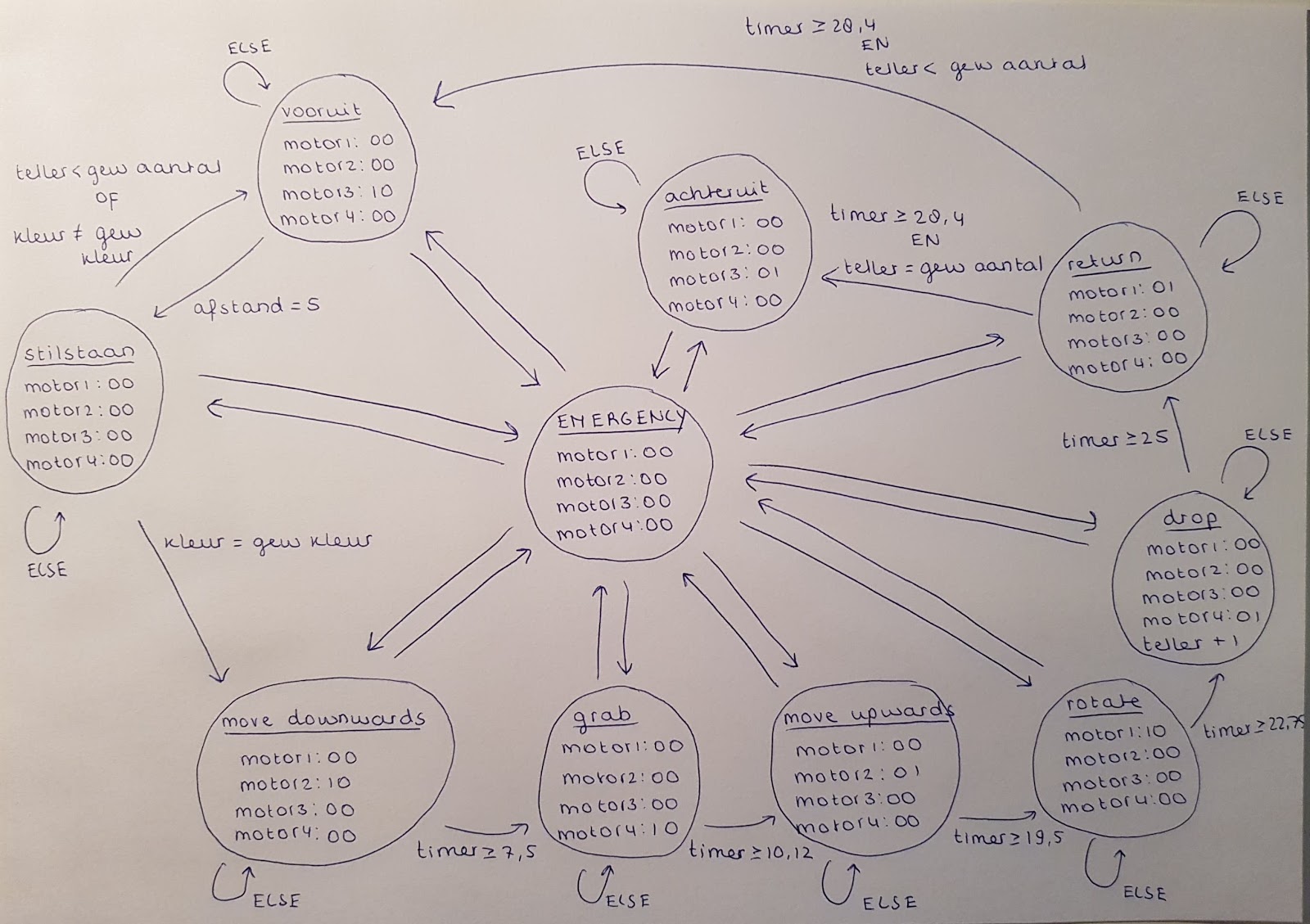
De eerste motor zorgt voor de rotatie van het robotje. Deze motor zal geactiveerd worden wanneer de grijparm met het blokje volledig bovenaan staat. Door middel van rotatie kan de grijparm dan de nodige draaibeweging uitvoeren om het blokje boven het bakje te laten vallen.

De tweede motor wordt gebruikt om de grijparm die de blokken vastgrijpt verticaal te laten bewegen. Nadat de kleurensensor bepaalt heeft welke kleur het blokje is zal het nagaan of het blokje moet worden opgepakt. Indien het moet worden verzameld zal de motor van de grijparm naar beneden bewegen totdat de grijparm helemaal beneden is. Als het blokje vastgegrepen is zal deze motor weer aangestuurd worden, maar deze keer in de andere richting zodat de grijparm naar boven gaat.

De derde motor wordt gebruikt om het hele toestel naar voor of naar achter te laten rijden. Eens dat de input is ingegeven en we op start drukken zal de robot autonoom naar voren rijden totdat de afstandssensor aangeeft dat er een blokje aanwezig is. Vervolgens stopt de motor met werken en zal de robot dus ook stoppen met rijden. Als het blokje vastgepakt en gedeponeerd is zal de motor terug aangestuurd worden om naar het volgende blokje te rijden. Dit proces wordt herhaald totdat de robot al de blokjes verzameld heeft. Hierna zal de motor in tegengestelde richting draaien waardoor het toestel achteruit zal manoeuvreren.

De laatste motor heeft als functie het blokje te grijpen. Aan het einde van de grijparm bevindt er zich een grijper. Als de grijparm volledig beneden is zal deze grijper beginnen met sluiten en zo een druk zetten op het blokje zodat deze stevig vast zit tussenin de grijper. Nadat de grijparm terug naar boven bewogen is en geroteerd heeft tot de stand waar het blokje moet vallen, zal deze grijper terug openen door de motor in de omgekeerde richting te laten aansturen.

### FSM



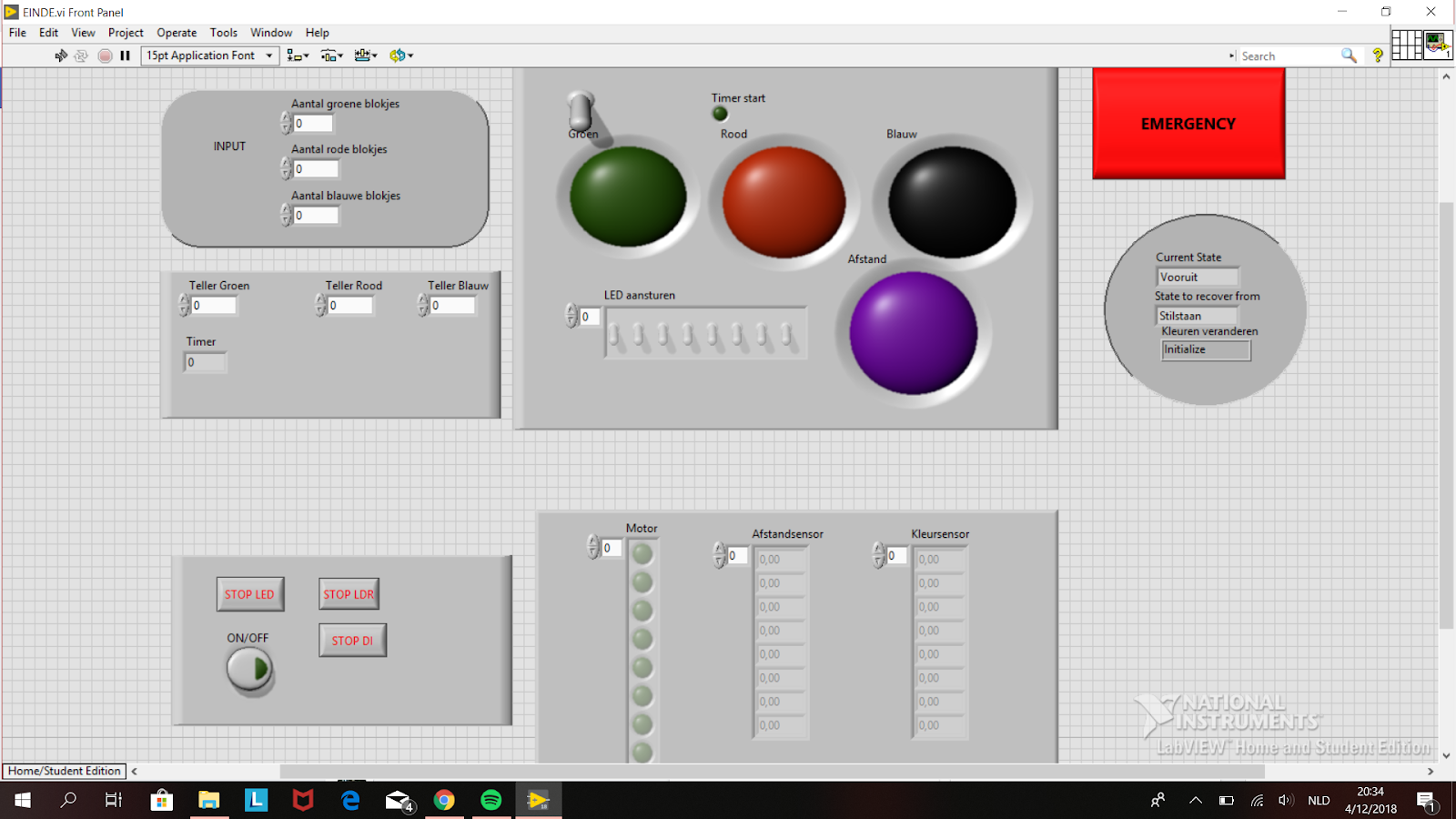
De eerste state waarmee de FSM begint is ‘stilstaan’. Wanneer de gebruiker de gewenste inputs heeft ingegeven en op de startknop drukt dan zal de robot vooruit bewegen. Aan de hand van een teller houdt de robot het aantal opgepikte blokjes bij. Zolang deze teller kleiner is dan het gewenste aantal blokjes zal de robot blijven vooruitrijden totdat de afstand tot het volgende blokje 4 cm bedraagt. Indien de kleur van dit blokje overeenstemt met de kleur van de input, dan zal de grijper naar beneden bewegen (‘move downwards’). Is dit niet het geval, dan zal de robot gewoon verder vooruitrijden tot het volgende blokje.

Nadat de grijper naar beneden bewogen heeft, zal het gaan grijpen (‘grab’), naar boven bewegen (‘move upwards’), met de klok mee draaien (‘rotate’), het blokje laten vallen (‘drop’) en weer terugdraaien (‘return’). Dit proces wordt bijgehouden met behulp van een teller. Wanneer alle tellers voor elke kleur blokje corresponderen met de gegeven input zal de robot achteruitrijden naar het startpunt.

Ook is er een noodknop (‘emergency’) aanwezig die ervoor zorgt dat het proces wordt gepauzeerd wanneer deze wordt ingedrukt. Wanneer de knop dan weer wordt losgelaten zal de robot doorgaan met de state waarin het zich bevond voordat de noodknop werd ingedrukt.

### Programma

Om ons programma zo logisch mogelijk te laten werken hebben we het opgedeeld in verschillende segmenten. Zo hebben we deel per deel geprogrammeerd om dan uiteindelijk alles samen te voegen en de verschillende ‘states’ op elkaar te laten afspelen.



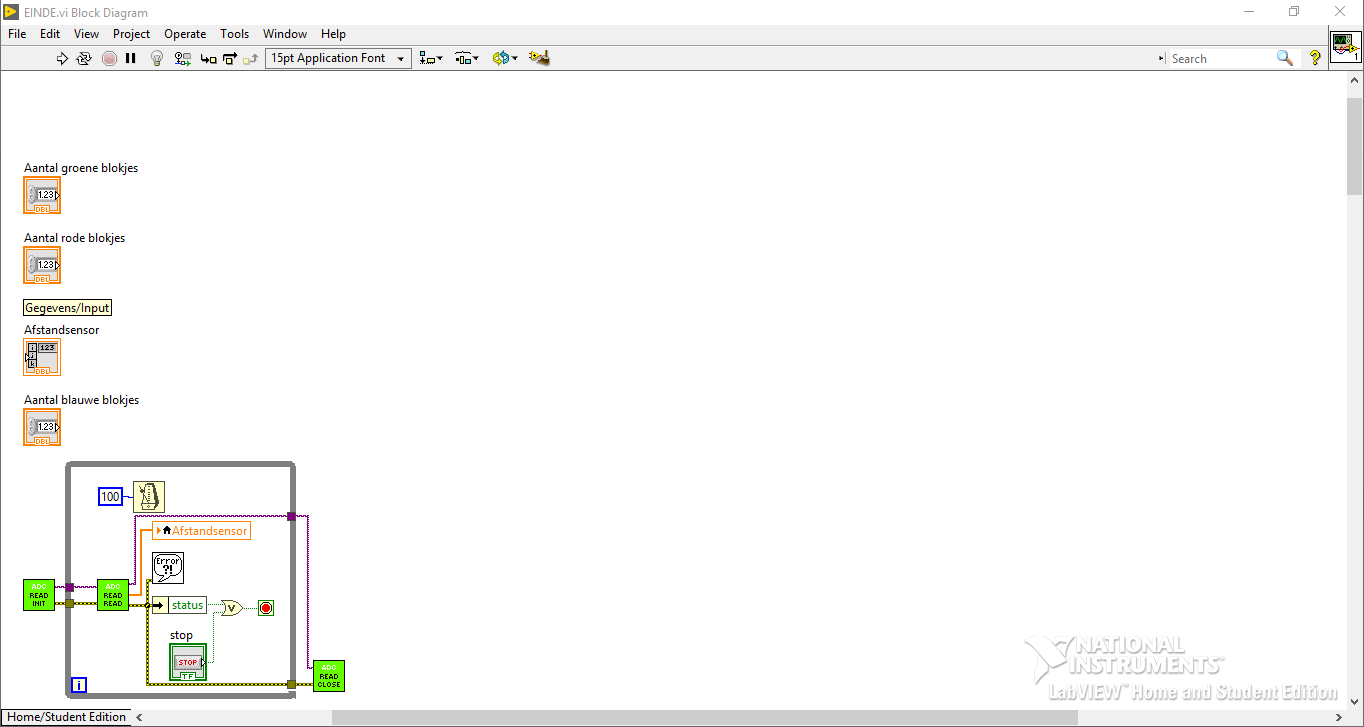
Beginnend met het frontpaneel, merkt men meteen enkele dingen op. Eerst en vooral de grote rode noodknop: deze is zo geprogrammeerd dat, indien men deze indrukt, het programma (en daardoor het robotje) stopt, ongeacht in welke state het zich bevindt.

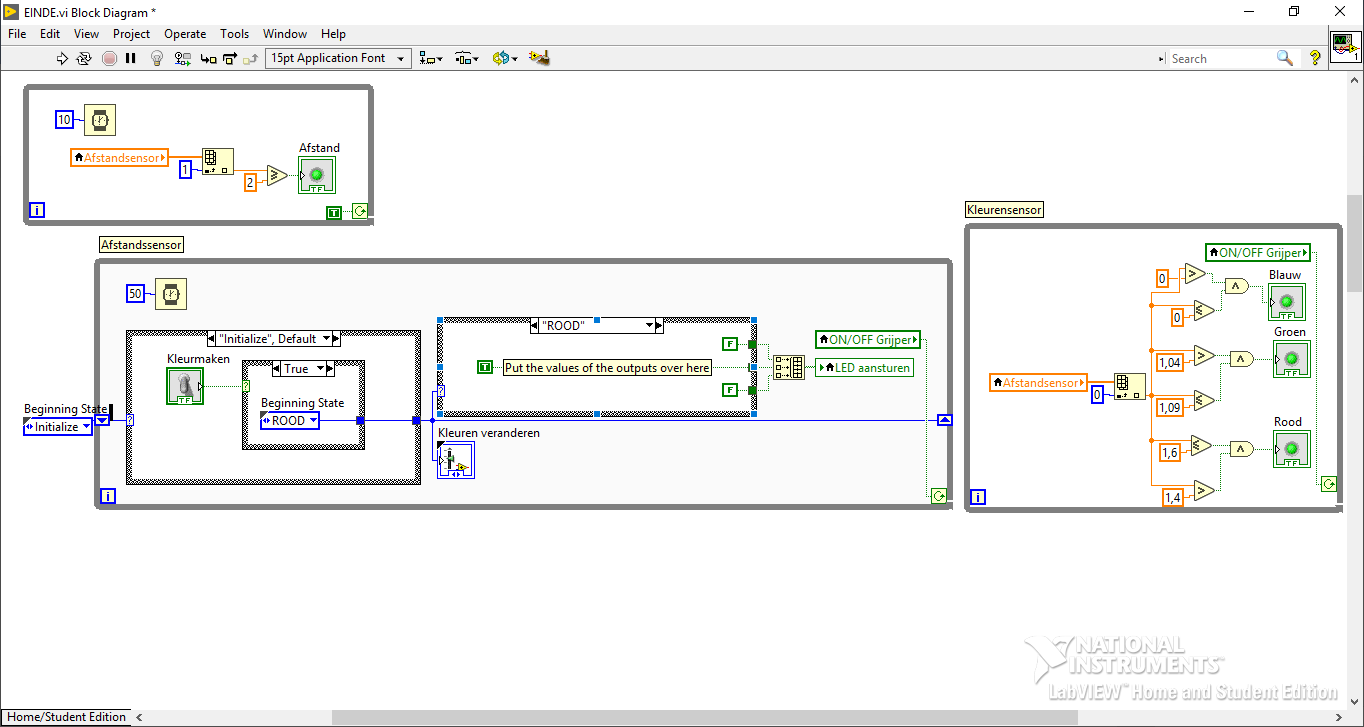
Een volgend item dat men meteen opmerkt zijn de drie LED-lichten. Deze dienen om aan te tonen welke kleur bolkje er zich momenteel voor het robotje bevindt. Indien er een groen blokje ligt dan zal de groene LED aanspringen, indien rood zal de rode LED branden, en hetzelfde voor de blauwe LED. Naast de drie hoofdLEDs merkt men nog een paarse LED. Deze geeft aan of er wel degelijk een object aanwezig is. Dit wordt bepaald door middel van een afstandssensor die de LED zal aansturen bij een specifieke waarde.

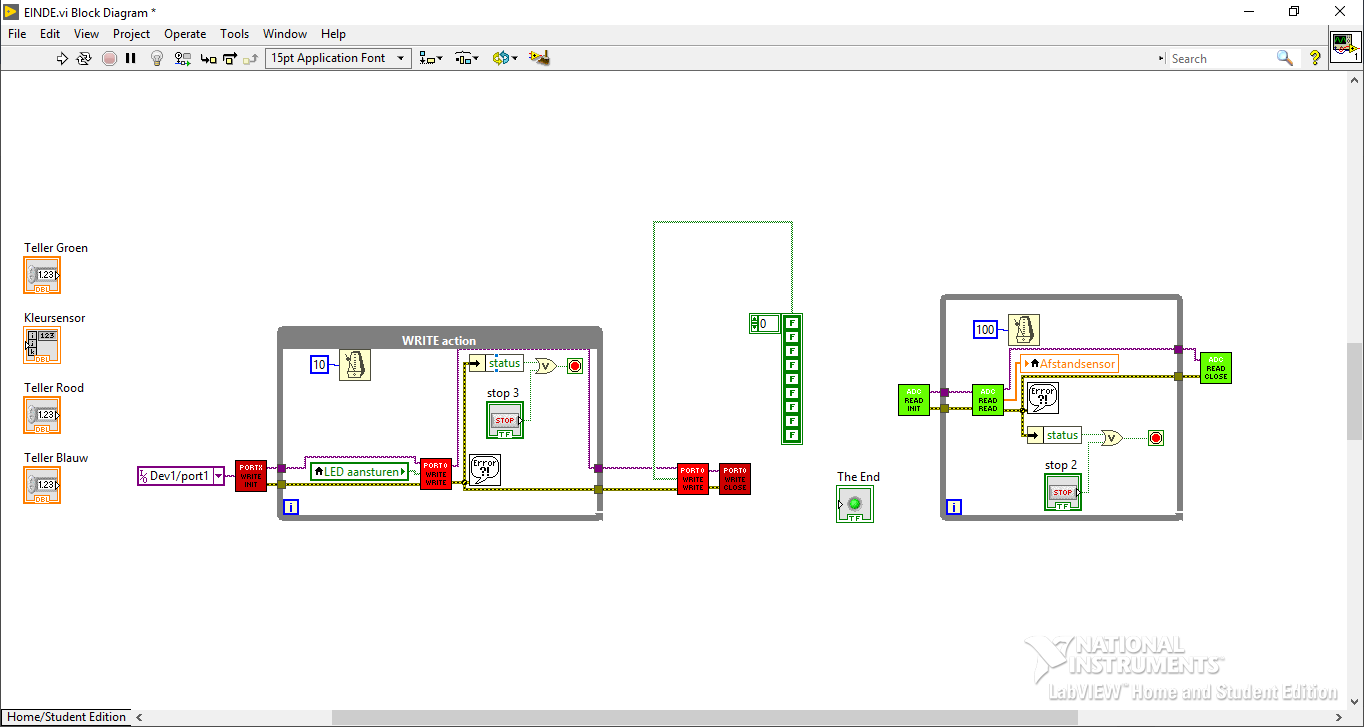
Rechts van de LEDs en onder de noodknop ziet men een bubbel met drie string indicatoren. De eerste geeft weer in welke staat de robot zich op een gegeven moment bevindt. Veronderstel dat de robot bijvoorbeeld vooruit aan het rijden is, dan zal dit hier af te lezen zijn. Onder de ‘current state’ bevindt zich een andere string indicator met de naam ‘state to recover from’. Deze slaat de ‘current state’ op en zorgt ervoor dat deze kan gelezen worden nadat er gebruik werd gemaakt van de noodknop zodat het programma gewoon kan verder gaan waar het gestopt is. De laatste string indicator geeft weer in welke staat de verschilende LEDs zich bevinden.

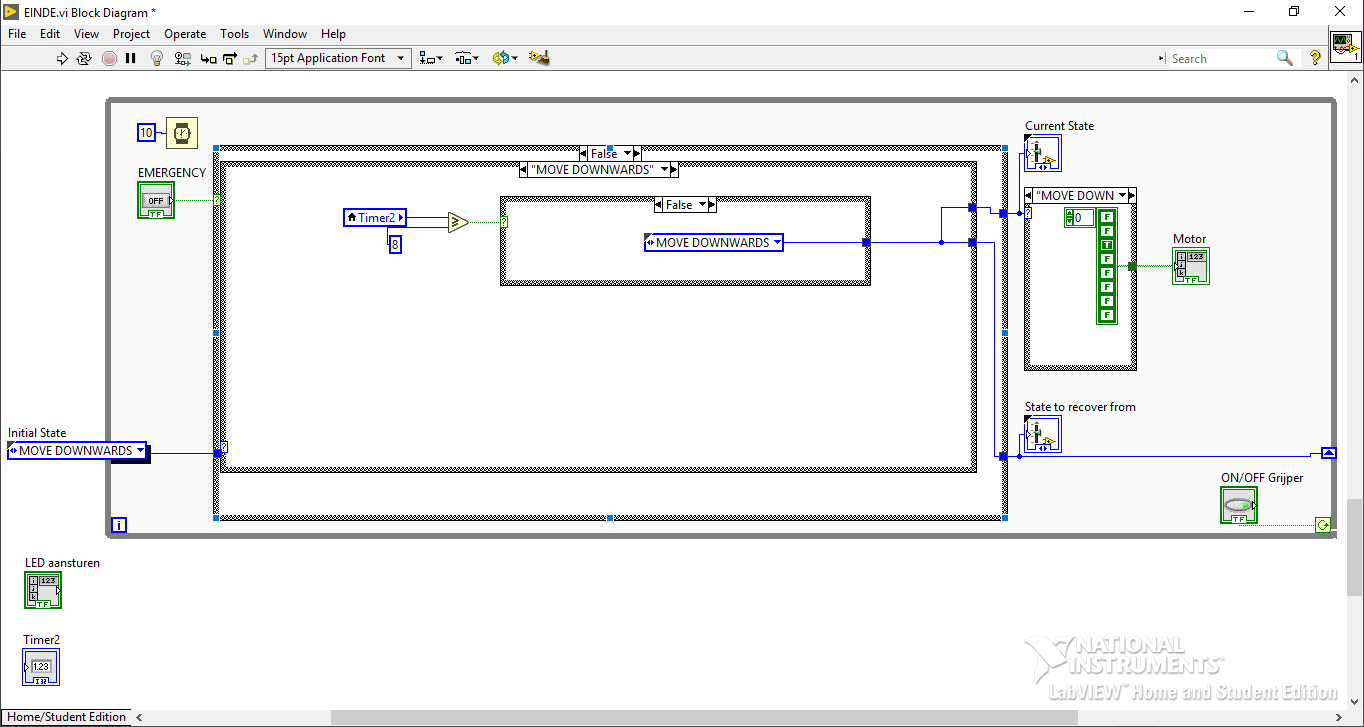
Verder ziet men nog acht andere LEDjes. Deze geven de status (aan/uit) van de motoren weer. Indien een motor aan staat dan zal het corresponderende LEDje branden, zo niet, dan blijft het LEDje uit. Deze LEDjes zullen veranderen gedurende de proef aangezien de gebruikte motoren van de state afhangen.

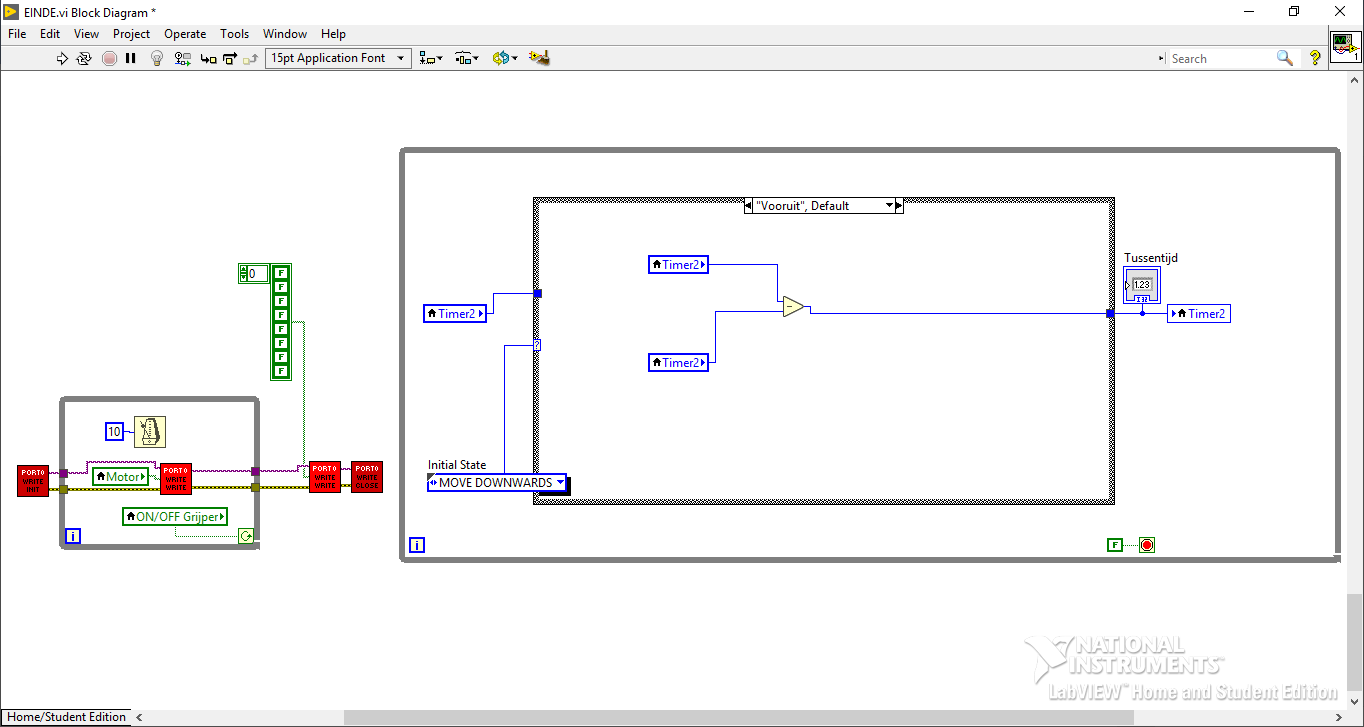
Als laatste merkt men nog een control en indicator voor het aantal blokjes. Aan de hand van de control kan de gebruiker aangeven hoeveel blokjes en in welke kleur hij wenst. Indien de robot dan een blokje van een bepaalde kleur heeft opgepakt dan zal dit in het programma te zien zijn aan de hand van de indicator.









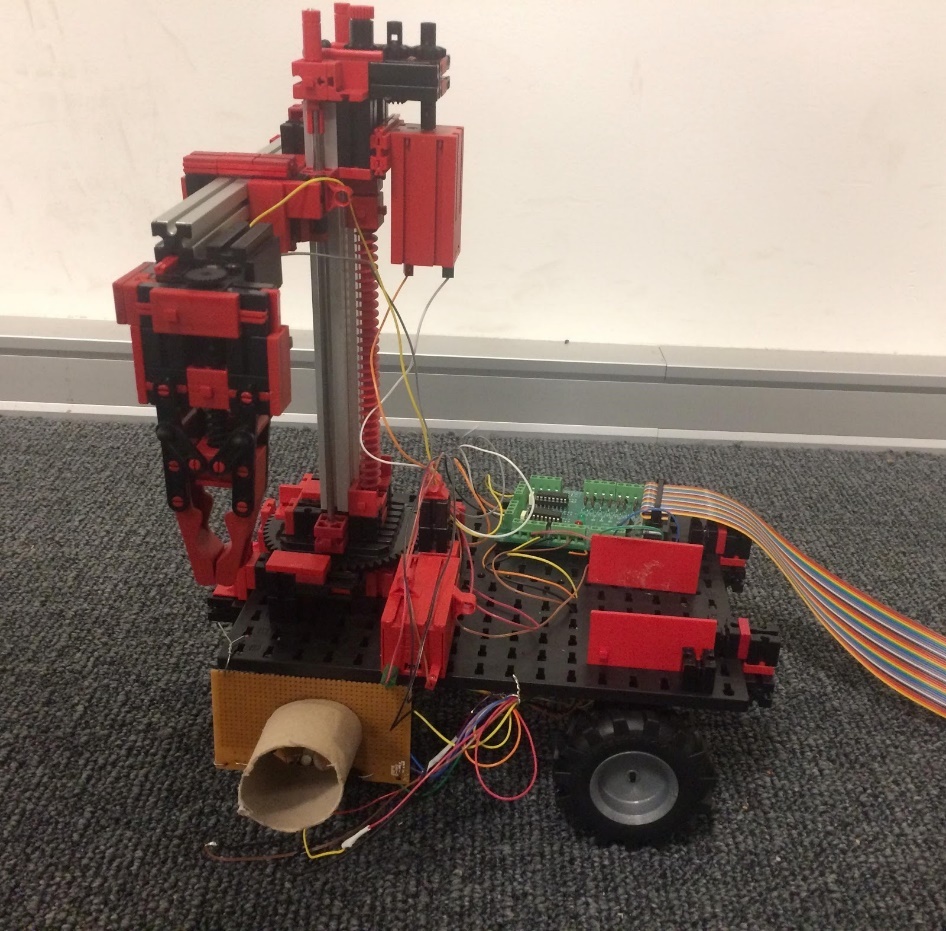


Al deze functies zijn terug te vinden in het blokdiagram van het programma en vereisen geen verdere specifieke uitleg. Wel worden er nog enkele algemene gebruiken en keuzes in het blokdiagram verder toegelicht.

Eerst en vooral hebben we uitbundig gebruik gemaakt van lokale variabelen. Deze zorgen ervoor dat men maar op een plaats deze variabele kan veranderen en op verschillende andere plaatsen dezelfde opgeslagen variabele kan gebruiken om naar te schrijven of af te lezen.

Verder hebben we geopteerd voor een teller in plaats van gebruik te maken van schakelaars. Dit omdat we niet wisten hoe de schakelaars nauwkeurig te gebruiken en we dit niet binnen de tijd konden uitdokteren. De teller werkt zo dat ze na een bepaalde cyclus wordt gereset tot 0 zodat ze de cylcus meerdere keren kan doorlopen.

2.3.7 Prototype



## Testen

Eén motor functioneerde niet naar behoren, waardoor we het programma in LabVIEW niet volledig konden testen. Het hele programma is zodanig opgebouwd dat de motoren afhankelijk zijn van elkaar, dus indien er één niet werkt kunnen we het programma niet volledig testen. Wel, konden we het programma testen totdat de niet-werkende motor aan de beurt kwam. Tot dat moment werkte de robot grotendeels zoals gewenst. De kleursensor werkt echter nog niet nauwkeurig genoeg om de juiste kleuren te kunnen detecteren, waarschijnlijk doordat de koker om de sensor niet donker genoeg is. Daarbovenop werkte de geïmplementeerde timer voor het grijpproces niet helemaal, maar na enkele aanpassingen was dit opgelost.

# Enterprising gedeelte

## Kostprijsberekening

Wij hebben ons zeven weken beziggehouden met dit project, maar indien men dit als fulltime job doet en hier iedere dag aan kan werken, zal de periode heel wat korter zijn. We schatten dat men binnen een maand een werkend concept zou kunnen hebben. Daarom gaan we bij deze berekening uit van een maand (31 dagen), met daaraan voorafgaand semesterlange cursus over LabVIEW. Bij de verdeling van de kosten zijn we uitgegaan van de onderdelen die wij hebben gebruikt bij onze eigen robot. Verder zijn we bij de huur uitgegaan van een kleine ruimte die drie ingenieurs nodig zouden hebben om aan de robot te kunnen werken en de verschillende onderdelen ervan in op te slaan. Hieronder worden de verscheidende projectfasen met hun kostenverdeling weergegeven.

### Opleidingen

De elektronica-ICT-ingenieur heeft vast al ettelijke ervaring met LabVIEW, maar om het nodige bij te leren is er nog een opleiding van 300 euro per persoon vereist. Deze opleiding is een zesmaandelijkse onlinecursus waarbij de ingenieur zelf kan bepalen wanneer hij deze volgt.

### Planning

Een van de ingenieurs kan de planning voor de maand opmaken, wat niet langer dan een dag zou moeten innemen.

* Loon

### Conceptgeneratie

Alle ingenieurs werken samen aan het brainstormen en kiezen van een concept. Dit concept verwachten ze klaar te hebben binnen 3 dagen.

* Loon
* Huur

### Constructiefase – programmeren

De elektronica-ICT-ingenieur werkt aan een werkend programma voor het bedachte concept. De ingenieur krijgt hiervoor tien dagen.

* Loon
* Huur
* LED
* LDR
* Afstandssensor
* Drie transistoren
* Zeven weerstanden
* NI USB 6008
* We rekenen de aankoop van de professionele versie van LabVIEW er niet bij aangezien we ervan uitgaan dat het ingenieursbedrijf hier al van in bezit is.

### Constructiefase – mechanisme

De mechanica ingenieur bouwt met de Fischertechnik een apparaat naar het behoren van de klant en krijgt hier net zoals de elektronica-ICT-ingenieur tien dagen voor.

* Loon
* Huur
* Fischertechnik
* Wielen

### Alles samenbrengen

Nadat er een werkend programma ontworpen en het apparaat gebouwd is kunnen deze twee onderdelen samengevoegd worden. Dit zou niet langer dan drie dagen mogen duren.

* Loon
* Huur

### Testfase

Alles is samengebracht en moet zorgvuldig getest worden. Hiervoor worden er vier dagen gerekend.

* Loon
* Huur

### Verbeterfase

Na het testen zullen er zaken zijn die niet werken of die verbeterd kunnen worden naar de wensen van de klant. Omdat na de testfase eventueel dingen herdacht en veranderd moeten worden, is het gepermitteerd dat deze fase wat langer zal duren. Daarom wordt er hier ook tien dagen de tijd voor genomen.

* Loon
* Huur

### Totaal

|  |  |
| --- | --- |
| Directe kosten | Euro (€) |
| Loon voor drie ingenieurs | 7800 |
| Afstandssensor | 21 |
| Fischertechnik | 210 |
| Wielen | 10 |
| LED | 3,50 |
| LDR | 0,20 |
| Transistoren | 0,12 |
| Weerstanden | 0,70 |
| NI USB 6008 | 145 |

|  |  |
| --- | --- |
| (In)directe kosten | Euro (€) |
| Huur | 500 |
| Opleiding | 300 |

Eigenlijk kunnen alle kosten als direct worden beschouwd, aangezien men precies kan zeggen voor welk product deze kosten zijn opgesteld. Dit komt doordat we aan stukproductie doen waarbij we rekening houden met de wensen van de individuele afnemer. Als men dit concept produceert aan de hand van massaproductie dan zullen bepaalde kosten wel als indirecte kosten geteld moeten worden.

De totale kostprijs komt neer op €8990,52. Aan de hand van de kostprijs kan de verkoopsprijs berekend worden via volgende formule: kostprijs + nettowinst = verkoopsprijs. Stel we werken hier met een winstmarge van 20%, dan wordt de verkoopsprijs: €8990,52 + €2247,63 = €11.238,15.

Hetzelfde resultaat zou eventueel kunnen behaald worden met een minder aantal ingenieurs, wat ervoor zorgt de kosten ook zouden zakken. Het grootste deel van de totale kostprijs naast de huur zijn immers de loonkosten. Bovendien zijn we ervan overtuigd dat ons project in sneller dan een maand kan gerealiseerd worden.

## Motovatiecurves en reflectie

### Arno Vermeir

#### Bespreking van de curve

“In het begin van het project was de motivatie heel hoog, dit is het geval bij elk nieuw project. Deze motivatie was wel snel verdwenen in de volgende weken door veel testen die ik moest voorbereiden en omdat ik een week heel ziek was. Ook het feit dat we toen inzagen hoeveel er moest gebeuren en in zo weinig tijd deed de motivatie zeker vallen.

Na al de testen gedaan waren (na week zes) heb ik de moed terug opgeraapt en begonnen aan een moeilijk deel van het systeem, de sensoren. Dit was een moeilijke taak en heeft zeer lang geduurd maar uiteindelijk was het gelukt en was de motivatie terug hoog.

Vanaf week negen begon de motivatie terug te dalen door de vele testen en de deadline die in zicht kwam.”

#### Taken en aantal uren

“In totaal heb ik 74 uur aan dit project gewerkt, dit is met elke les EE3 erbij, dit betekent ook dat de lessen LabVIEW bij deze uren zitten. Voor de rest heb ik zeker het meeste tijd (samen met Yoni) aan de sensoren bezig geweest.”

#### Appreciatiepunt

“Als appreciatiepunt voor EE3 geef ik een 7/10, het was een goede manier om zelfstudie te beheersen, en nog veel andere dingen te leren. Ik had wel het gevoel dat ik na de lessen van Labview nog niet in staat was om het hele systeem van de robot te programmeren.”

### Ciaran Van Hoeserlande

#### Bespreking van de curve

”Net zoals de rest van het team was ik bij aanvang van het projectwerk enthousiast aangezien ik geen idee wat er me te wachten stond (en aangezien ik van nature nieuwsgierig en avontuurlijk ben aangelegd gaf dit mij een wel wat motivatie). De introductiecursussen over LabVIEW waren wel boeiend, maar vanaf dat we deze kennis plots zelfstandig moesten toepassen op onze robot daalde mijn motivatie. Ik had geen idee wat of hoe en de docenten hielpen daar eerlijk gezegd ook niet echt bij. De eerste deadline was voor mij een dieptepunt, aangezien niets was gelukt en ik daardoor dacht dat het gehele project een flop zou worden. De daaropvolgende weken zag ik een toename in mijn motivatie aangezien ik toch niet bij de eerste tegenslag wou opgeven en ik het project zelf nog steeds interessant vond om aan te werken. Toen ik mijn eerste FSM volledig zelfstandig hebben kunnen programmeren voelde ik me euforisch en kreeg ik weer heel wat vertrouwen (in week 9). De laatste weken daalde m’n motivatie echter weer doordat we gedurende die periode alles moesten samenbrengen en ik niet zoveel vertrouwen had dat het zou lukken. Toch heb ik op het allerlaatste moment nog veel motivatie gekregen doordat het ons uiteindelijk toch wel gelukt is.”

#### Taken en aantal uren

“Ik schat zo’n 75 uur aan het project te hebben gewerkt.”

* Bouwen (6 uur)
* Programmeren (12 uur)
* Verslag (9 uur)
* Lesuren (48 uur)

#### Wat heb ik geleerd?

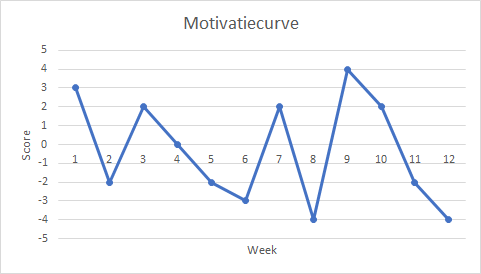
“Een van de grootste zaken dat ik heb geleerd is dat ik meer moet kunnen vertrouwen op mijn teamleden (dus teamwerk). Ik ben van nature perfectionistisch en dat maakt het dus niet altijd even gemakkelijk om andere mensen met een taak toe te vertrouwen aangezien ik ben bang dat hun uitvoering ervan niet aan mij verwachtingen zal voldoen. Ook heb ik geleerd te programmeren in LabVIEW. Ik vind het echter niet zo voor de hand liggend als LabVIEW, gegeven de context waarin we het hebben moeten leren. Als laatste heb ik ook leren beter doelgericht te plannen dankzij de verschillende sprints. Door mezelf verantwoordelijk te stellen over een taak was ik automatisch sneller geneigd om deze tijdelijk en meteen volledig uit te voeren.”

#### Appreciatiepunt

“Als ik terugblik op de zeven weken dat ik aan het project heb gezeten moet ik wel zeggen dat ik vrij tevreden ben over het resultaat en het vak. Een zaak dat ik meteen waardeerde is de vrijheid in het project. We mochten zelf bepalen wat we zouden in elkaar steken en hoe we dat konden realiseren. Aan de andere kant hadden we dan wel niet zoveel besef van wat er ons te wachten stond en hoe realistisch ons concept was. Zoals eerder vermeld vond ik persoonlijk de docenten ook niet zo behulpzaam; ze lieten blijken alsof alles basiskennis was. En de beperkte informatie op het internet maakte het er zeker niet gemakkelijker op. Als ik het vak een punt zou moeten geven, dan zou ik het een 7,2 op 10 scoren.”

### Connor Verhulst

#### Bespreking van de curve



“Bij het begin van het project had ik veel zin om eraan te beginnen en had ik dus een goede dosis motivatie. De negatieve uitschieters van de curve komen door de deadlines, omdat deze veel stress met zich meebrachten en mijn motivatie dus negatief beïnvloeden. Ook de test van EE3 deed mijn motivatie dalen, omdat ik daar veel tijd in moest steken om te leren. In week 9 had ik een motivatieboost, omdat de LabVIEW test van week 8 goed was verlopen en we eindelijk goede vooruitgang boekten met de robot. In week 12 is de demonstratie en heb ik nog 3 andere testen, dus gaat mijn motivatie hoogstwaarschijnlijk zeer laag zijn.”

#### Taken en aantal uren

* Bouwen: samen met Demi de robot bouwen (4 uur).
* Wielen: samen met Demi heb ik de voorgaande wielen (niet de wielen die er nu onder zitten) gebouwd en geprogrammeerd zodat het robotje kon rijden (5 uur).
* Verslag (3 uur)
* Eindprogramma: Eerst heb ik alle FSM’s bij elkaar gezet. Daarna heb ik meegeholpen om het uiteindelijke programma effectief te doen werken aangezien er nog veel errors waren (7 uur).
* Lesuren (48 uur)

#### Wat heb ik geleerd?

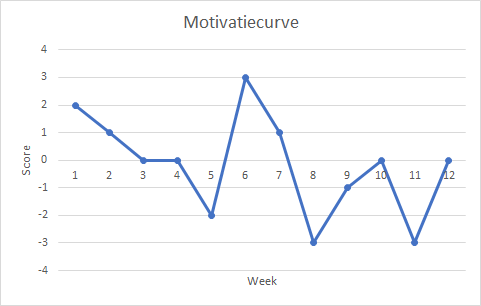
“Uit dit project heb ik vooral beter leren samenwerken met anderen. Bij de conceptgeneratie was het belangrijk dat we naar elkaars ideeën luisterden en samen een concept kozen waar we allemaal tevreden over waren. Het meeste heb ik natuurlijk bijgeleerd over LabVIEW, omdat ik hier voor EE3 nog nooit mee had gewerkt. Probleemoplossend te werk gaan is ook een vaardigheid die ik verbeterd heb bij dit project aangezien we veel obstakels zijn tegengekomen die we zelf moesten oplossen. Ten slotte heb ik geleerd om efficiënter en meer doelgericht te werken. Door de deadlines en de verschillende afzonderlijke taken leer je gerichter te werk te gaan zodat het geheel op tijd tot een goed einde kan komen.”

#### Appreciatiepunt

“Het feit dat we zelf creatief te werk mochten gaan vond ik het leukste aan dit project. De afwezigheid van strikte regels werkten voor mij motiverend om het project zelf zo goed mogelijk te doen. Met labview werken was soms moeilijker, omdat we veel errors kregen bij de eerste pogingen. Eens ik meer ervaring had opgedaan en het programma beter beheerste vond ik het echter wel leuk om mee te werken. Het negatieve aan EE3 vond ik dat we in het begin met relatief weinig info aan het werk moesten, waardoor we vaak heel lang vastliepen bij bepaalde problemen twijfels die we over het project hadden. Algemeen vond ik EE3 wel leuker dan EE2 en geef ik het een score van 7/10.”

### Demi Alarm

#### Bespreking van de curve



“De eerste week was ik nog zeer gemotiveerd, omdat ik zin had om zelf iets te gaan bouwen. Daarna loopt de lijn wat naar beneden, wat komt door de lessen LabVIEW. Ik vind dit programma niet zo fijn om mee te werken dus dit maakte mij iets minder gemotiveerd. De hoge uitschieter naar boven was doordat de eerste scrum zeer goed verliep. Hierdoor kreeg ik zin om er verder aan te werken en om het robotje nog meer functies te kunnen geven. In week 8 was de LabVIEW test wat niet zo goed ging en dus ook te zien is in mijn motivatiecurve.  Na deze test kwamen de deadlines en was het wat stressen met de groep om de robot en het verslag op tijd af te krijgen. Hierdoor was mijn motivatie niet heel hoog en werkte ik er met veel tegenzin aan. Aan het einde kreeg ik toch nog wat motivatie, omdat het leuk was om te zien wat de robot al allemaal kon en ik graag de robot helemaal af wilde maken om het doel te bereiken dat we aan het begin van het project in gesteld hadden.”

#### Taken en aantal uren

* Bouwen: samen met Connor heb ik het robotje gebouwd (4 uur).
* Wielen: samen met Connor heb ik de voorgaande wielen (niet de wielen die er nu onder zitten) gebouwd en geprogrammeerd zodat het robotje kon rijden (5 uur).
* Verslag (5 uur)
* Eindprogramma: Alle FSM’s moesten bij elkaar gezet worden. Nadat ze bij elkaar waren gezet werkte het nog niet en heb ik meegeholpen om het eindprogramma werkend te maken. Hierna hebben we nog getest (6 uur).
* Lesuren (48 uur)

#### Wat heb ik geleerd?

“Zoals bij vorige groepsprojecten heb ik geleerd dat samenwerken niet altijd makkelijk is, omdat iedereen verschillende ideeën heeft over hoe iets moet gaan werken en hoe iets moet komen uit te zien. Wat belangrijk is bij deze gevallen is om naar elkaar te luisteren en de potenties van andere ideeën in te zien, maar ook je eigen inzichten te tonen aan de andere leden van je team.

Ik heb ook leren werken met LabVIEW, evenals het overzetten van een programma in LabVIEW naar een werkend robotje.

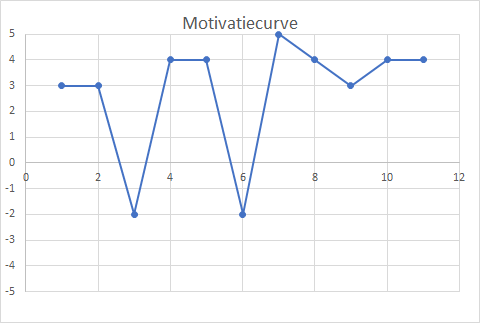
Communiceren met elkaar is belangrijk als het gaat om planning en verdeling van de taken. Bij ons was het vaak nogal lastig wanneer iemand de doos nodig had, maar de doos dan net bij iemand anders op kot stond. Dit had voorkomen kunnen worden door beter te plannen.”

#### Appreciatiepunt

“Wat ik vooral apprecieer is dat je zoveel vrijheid krijgt in het maken van je concept. Op deze manier kun je wat creatiever zijn dan wanneer je je aan heel veel regels moet houden zoals bijvoorbeeld bij EE2. Wat ik wel jammer vind is dat er vrij weinig uitleg wordt gegeven, maar uiteindelijk was het wel duidelijk dat we zelf gewoon veel met LabVIEW moesten oefenen en dat we veel zelf moesten uitzoeken. Dit werkte allemaal wel, maar wat meer ondersteuning zou fijn zijn. LabVIEW vond ik absoluut geen fijn programma om mee te werken en aangezien je grotendeels hiermee bezig bent is dit wel jammer. Al bij al geef ik EE3 een 6,7/10.”

### Yoni Belis

#### Bespreking van de curve



“De lage punten kwamen vooral doordat we in deze weken problemen hadden en dingen niet lukte. De week erna zijn dan meestal hogere punten omdat uiteindelijk het dan toch wou werken. Bijvoorbeeld toen we de kleurensensor moesten maken hebben we veel tijd ingestoken en liep er veel fout waardoor mijn motivatie minder was maar toen het uiteindelijk werkte was mijn motivatie wel terug hoog. Als je mijn curve vergelijkt met de gemeenschappelijke curve dan zie je dat het grotendeels wel overeenkomt maar in de laatste weken mijn motivatie hoger ligt dan de gemeenschappelijke.”

#### Taken en aantal uren

“Als eerste heb ik samen met Ciaran en Arno de eerste FSM gemaakt (4 uur). Dit was de FSM die de basisprincipes van de robot uitvoerde. Als tweede heb ik samen met Arno de kleurensensor gemaakt (LDR +RGB). Dit heeft in totaal ongeveer 20 uur geduurd omdat 5 uur hiervan we een error hadden die te maken bleek te hebben met de DAQ-drivers. Nadat we van pc waren gewisseld werkte het programma namelijk perfect. Voor derde taak heb ik alleen de afstand sensor geprogrammeerd. Dit ging al veel sneller omdat ik hier geen errors meer kreeg en was ongeveer 3 uur voor het programmeren en vastmaken. Als laatste hebben we alle programma’s bij een gezet maar omdat hier nog vrij veel fouten in zaten heb ik samen met Demi de FSM veranderd zodat deze wel werkte (6 uur). Nadien heb ik ook nog een paar fouten uit het programma gehaald (30 min). Als laatste zijn er dan ook nog de uren van de sessies zelf (48) en de uren aan het verslag (2 uur).

Wat ik vooral geleerd heb van EE3 is dat communicatie heel belangrijk is. Op sommige momenten liepen er binnen ons team paar dingen mis wegens slechte communicatie. Dit was dan vooral wanneer we de doos nodig hadden om te bouwen. Een andere punt is natuurlijk samenwerken met andere personen dit is niet altijd even makkelijk omdat soms wel andere ideeën hebben. Als laatste heb ik ook leren werken met LabVIEW en hoe je in dit programma een robot kan aansturen via sensoren en motoren.”

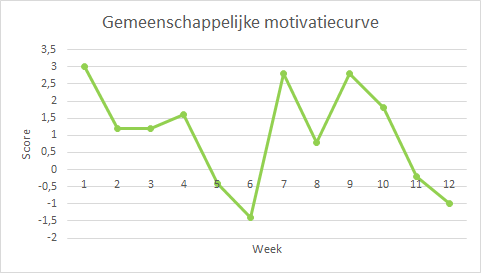
#### Wat heb ik geleerd?

“Wat ik vooral geleerd heb van EE3 is dat communicatie heel belangrijk is. Op sommige momenten liepen er binnen ons team paar dingen mis wegens slechte communicatie. Dit was dan vooral wanneer we de doos nodig hadden om te bouwen. Een andere punt is natuurlijk samenwerken met andere personen dit is niet altijd even makkelijk omdat soms wel andere ideeën hebben. Als laatste heb ik ook leren werken met LabVIEW en hoe je in dit programma een robot kan aansturen via sensoren en motoren.”

#### Appreciatiepunt

“Het is wel leuk dat je bij EE3 meer vrijheid krijgt dan bij EE2 bijvoorbeeld. Dit is leuker omdat je meer zelf kan denken en ontwerpen en niet vasthangt aan een vast eindresultaat zoals bij de pendulum van EE2. Deze vrijheid heeft echter ook wel nadelen want voor sommige onderwerpen hadden we vrij weinig informatie gekregen waardoor dingen moeilijker leken dan ze echter waren. Hierdoor hebben we wel vrij veel tijd verloren wat spijtig is. Algemeen vond ik EE3 wel leuker dan EE2 en geef ik wel een 7/10.”

### Bespreking van de gemeenschappelijke curve



#### Arno over algemene curve

“Over het algemeen ziet mijn curve er gelijkaardig uit met de gemiddelde curve, iedereen in onze groep had testen en hierdoor zal mijn curve overeenstemmen met die van de gemiddelde. Enkel in week 8 is er bij mij meer motivatie omdat toen de afstandssensor eindelijk af was en dit mijn motivatie na veel werk erg heeft opgekrikt.”

#### Ciaran over algemene curve

“Op het eerste zicht is mijn individuele motivatiecurve vrij gelijkend met de gemeenschappelijke, alhoewel dat ik algemeen bekeken een tikkeltje meer gemotiveerd was dan mijn team. In week 10 en 11 had ik het blijkbaar wel wat moeilijker dan mijn team, aangezien dat de weken waren waarin we alles moesten samenbrengen en dat dat voor mij de echte test van ons project was. Verder liggen mijn minima en maxima altijd wel eentje hoger dan de extremen van de gemeenschappelijke motivatiecurve.”

#### Connor over algemene curve

“Als je mijn curve vergelijkt met de gemeenschappelijke motivatiecurve zie je dat de vorm relatief goed overeenkomt, maar dat mijn hoogste en laagste waarden extremer zijn dan die van de gemeenschappelijke curve. Algemeen lag mijn motivatie volgens deze curves blijkbaar lager dan de motivatie van de rest van het team.”

#### Demi over algemene curve

“Als je mijn curve vergelijkt met de gemeenschappelijke motivatiecurve zie je dat de vorm relatief goed overeenkomt, maar dat mijn hoogste en laagste waarden extremer zijn dan die van de gemeenschappelijke curve. Algemeen lag mijn motivatie volgens deze curves blijkbaar lager dan de motivatie van de rest van het team.”

#### Yoni over algemene curve

“Als je mijn curve vergelijkt met de gemeenschappelijke curve dan zie je dat het grotendeels wel overeenkomt maar in de laatste weken mijn motivatie hoger ligt dan de gemeenschappelijke. Dit komt waarschijnlijk door de vele deadlines dat algemeen de motivatie lager ligt, maar doordat onze robot begint te werken ligt mijn motivatie juist hoger omdat ik het nu af wil krijgen.”

# Besluit

Algemeen kunnen we concluderen dat, hoewel we ons project succesvol hebben afgerond, er nog veel ruimte was voor verbetering.

De CBC zelf voldoet aan de vereisten en reageert gepast op de gewenste input. We hadden deze echter kunnen verbeterd door spraakherkenning toe te voegen. Hierdoor zou de CBC een extra tool hebben om de gebruiksvriendelijkheid te vergroten.

Ook op vlak van teamwerk en efficiëntie hadden we het beter kunnen aanpakken. Door betere afspraken te maken en vanaf het begin een teamleider aan te stellen zou het project een stuk vlotter zijn verlopen. Zowel tijd als kosten kunnen bespaard zijn geweest door betere communicatie en meer overleg binnen het team.

Ondanks de ruimte voor verbetering, hebben we als resultaat een werkende CBC die aan zijn productvereisten voldoet. Elk teamlid heeft zijn deel gedaan en door voldoende samenwerking hebben we het project succesvol kunnen beëindigen.

Het project heeft ons geleerd om creatief aan het werk te gaan en zelf de problemen op te lossen die we tegenkwamen om ons uiteindelijke doel te kunnen realiseren. De vrijheid om onze visie zelf tot leven te brengen apprecieerden we als team het meeste aan EE3, dit had een motiverend effect op onze groep. Uit de nieuwe inzichten op vlak van efficiëntie en samenwerking leiden we af dat het vak EE3 een goede invloed heeft gehad op onze vorming als ingenieurs in spe.

1. 00 en 11 corresponderen met dezelfde toestand, namens de ‘IDLE’-state [↑](#footnote-ref-1)