

AAN HET WERK MET PROFILAB EXPERT 4.0.

Niets uit deze geautomatiseerd enige wijze, hetzij microfilm of op welke toestemming van de

Ondanks alle aan de auteur geen aansprakelijkheid kunnen voortvloeien uit enige voortkomen.

Auteurswet 1912 bij het besluit van 20 besluit van 23 Augustus 1985, Stb 471 en de daarvoor wettelijke verschuldigde stichting Pro (postbus 725, 1180 AS overnemen van gedeelte(n) uit deze uitgave in ander compilatiewerken (artikel 1 auteurswet 1912) te wenden.

uitgave mag worden verveelvoudigd of opgeslagen in een gegevensbestand, of openbaar gemaakt in enige vorm of op elektronisch, mechanisch, door middel van druk, fotokopie, wijze dan ook, zonder voorafgaande schriftelijke auteur. Alle rechten voorbehouden.

samenstelling van dit boek bestede zorg kan de aanvaarden voor eventuele schade die zou fout die in deze uitgave zou kunnen

Juni 1974, Stb 351, zoals gewijzigd bij het artikel 17 auteurswet 1912, dient men vergoeding te voldoen aan de Amstelveen). Voor het bloemlezingen, readers en dient men zich tot de auteur



Copyright © 2016 H van Zwieten.

Auteur: H van Zwieten.

Druk: Pumbo.nl

Omslagontwerp: H van Zwieten.

Vormgeving binnenwerk: H van Zwieten.

Inhoudsopgave

Over de auteur	Blz.: 5
Voorwoord	Blz.: 6
Hoofdstuk 1: Over het programma Profilab expert 4.0	Blz.: 7
Hoofdstuk 2: Drukknoppen en leds	Blz.: 9
2.1: Schakelaars en poorten	Blz.: 12
2.2: Schakelaars en RGB	Blz.: 13
Hoofdstuk 3: Analoge elementen	Blz.: 14
3.1: Analoog en digitaal combineren_1	Blz.: 15
3.2: Analoog en digitaal combineren_2	Blz.: 17
Hoofdstuk 4: De K8055	Blz.: 18
4.1: Digitale uitgangen	Blz.: 19
4.2: Teller	Blz.: 21
4.3: Projector	Blz.: 23
4.4: Analoge ingang	Blz.: 24
4.5: Analoge uitgang	Blz.: 26
4.6: X/Y schrijver	Blz.: 28
4.7: Schakelen tussen potmeter print en front	Blz.: 29
4.8: Quiz tafel	Blz.: 31
4.9: Motorregelaar	Blz.: 33
4.10: Timer opkom vertraagt	Blz.: 36
4.11: Stappenmotorregeling	Blz.: 38
4.12: Licht donker schakeling	Blz.: 42
4.13: Data opslaan	Blz.: 44

4.14: Licht effecten	Blz.: 46
4.15: Programmeren met labels	Blz.: 48
Hoofdstuk 5: Over Pic controllers en interfaces	Blz.: 50
5.1: Controllerboard met 16F887	Blz.: 50
5.2: Print controllerboard	Blz.: 51
5.3: Seriële interface met PC817	Blz.: 52
5.4: Schema Seriële interface	Blz.: 52
5.5: print seriële interface	Blz.: 53
5.6: Over het programma Proton	Blz.: 53
5.7: Over de Pickit_2	Blz.: 56
Hoofdstuk 6: Werken met Profilab en Pic controllers	Blz.: 58
6.1: Het serieel binnen halen van een analog signaal	Blz.: 58
6.2: Het serieel uitsturen van een analog signaal	Blz.: 65
6.3: Het serieel aansturen van uitgangen	Blz.: 70
6.4: Het serieel binnen halen van ingangen	Blz.: 76
6.5: Het serieel besturen van een servo	Blz.: 85
6.6: Het serieel besturen van een dimmer	Blz.: 89
6.7: Het serieel besturen van een elektromotor	Blz.: 95
6.8: Relaischakeling met XBEE	Blz.: 109
Hoofdstuk 7: Werken met een digitale multimeter	Blz.: 123
7.1: Het serieel meten van spanning	Blz.: 125
7.2: Het serieel meten van stroom	Blz.: 127
7.3: Het serieel meten van temperatuur	Blz.: 129

Hoofdstuk 8: Een paar praktijk voorbeelden	Blz.: 132
8.1: Vijfde as voor een freesbank	Blz.: 132
8.2: Alarm melding voor een schip	Blz.: 137
8.3: Sprong test training	Blz.: 147
8.4: Handbal training	Blz.: 149
8.5: circuit training	Blz.: 152

Over de auteur

Vanaf mijn zesde jaar was ik al met techniek bezig, voornamelijk Mecano, (technisch lego was er toen nog niet) ik bouwde er van alles en nog wat van. Al snel kwamen er ruitenwissermotoren aan te pas, om alles automatisch te laten werken. Iets later kwamen de elektro bouwdozen van Philips aan bod, en daarna de bread boards met de losse componenten om met van alles en nog wat te experimenteren.

Op mijn twaalfde had ik mijn eerste motorfiets in elkaar geknutseld; van drie motorblokken had ik één werkend blok gemaakt (Jawa 250cc tweetakt) en een frame gemaakt van diverse frames. Heb jaren plezier gehad met dat ding, totdat hij op koninginnedag in beslag genomen werd door de politie.

In het voortgezet onderwijs heb ik de richting metaal en autotechniek gedaan en ben daarna als automonteur aan het werk gegaan. Dat heb ik een aantal jaren gedaan, en ben daarna toch overstapt op de elektrotechniek. Voornamelijk in de machinebouw, paneelbouw, en besturingstechniek.

Heb ook veel storingen gelopen bij allerlei bedrijven, voornamelijk in de industrie aan allerlei machines.

En zo ben ik na verdere studie in de elektronica gerold, met name in de ontwikkeling van allerlei interface printen en de ontwikkeling van besturingen, zowel via de pc als met controllers.

Opleiding

Metaal: Metaalbewerker; (Draaien, frezen, lassen).

Autotechniek: Automonteur; (Alle voorkomende werkzaamheden).

Elektrotechniek: Elektromonteur; (Paneelbouw, Machinebouw, Besturingstechniek).

Elektronica: Algemeen; (Digitaal, Analoog, Print ontwerp).

Besturingstechniek: Algemeen; (PLC, PC, Microcontrollers).

Meet en regeltechniek: Algemeen; (PLC, PC, Microcontrollers).

Hobby's

Elektronica, mechanica, programmeren van microcontrollers, programmeren in Profilab, modelbouw, printen ontwerpen en maken, technische boeken lezen, dus eigenlijk alles wat met techniek te maken heeft interesseert mij.

Voorwoord

Dit boek gaat over werken met Profilab Expert 4.0, een programma van ABACOM.

ABACOM www.abacom-online.de is een Duits ingenieurs bureau, welke dit programma op de markt heeft gebracht. Zelf werk ik al jaren met dit programma, tot grote tevredenheid.

Er is echter weinig te vinden over het werken met het programma en wat er allemaal mogelijk is. Misschien komt het wel door de enorme uitgebreidheid van het programma. Ik ga dan ook proberen het één en ander uit te leggen over het werken met Profilab.

We gaan eenvoudig van start. Profilab werkt met een grafische programmeer omgeving; je knoopt dus functie blokken en poorten aan elkaar om tot een werkend geheel te komen.

Het is de bedoeling om de voorbeelden te gaan maken met een K8055 van Velleman. Deze kaart (één van de vele) wordt ondersteund door Profilab en kan via de usb poort aangesloten worden op de pc.

Maar eerst maken we wat simpele voorbeelden met wat schakelaars en drukknoppen om een beetje aan het programma te wennen.

Later wil ik ook wat voorbeelden maken met een PIC microcontroller 16F887 die kan communiceren met Profilab via de usb poort (met een usb naar serieel converter) of via de seriële compoort van de pc.

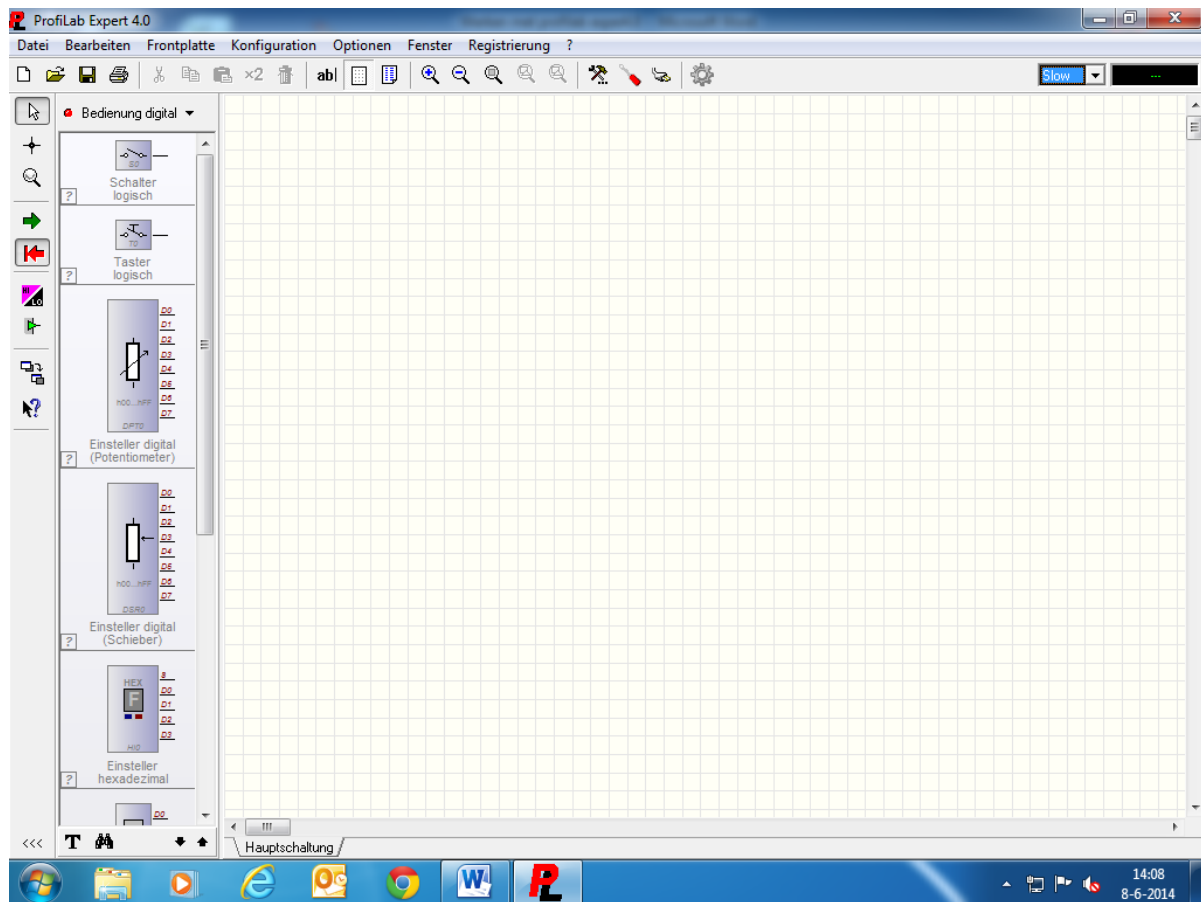
Ook zal ik wat schema's en printen plaatsen over de onderwerpen waar we het over gaan hebben.

Profilab draait zowel onder Windows XP, 7, en 8. Zelf heb ik 7 en daar worden ook de voorbeelden in gemaakt.

Profilab installeert automatisch; dit zal dus niet voor problemen zorgen. Het kan in een paar talen geïnstalleerd worden waaronder Duits. Daar heb ik destijds ook voor gekozen en ben daar eigenlijk nooit van afgestapt. Maar wie het in het Engels of Frans wil installeren is daar natuurlijk vrij in. Ik ga dus ook in het begin de Duitse termen van componenten gebruiken; later in deze uitleg moet iedereen de plaats van de componenten wel weten te vinden. Dan gaan we nu maar eens van start met het programma, hoop dat velen er net zoveel plezier aan beleven als ik.

Veel lees- en programmeerplezier.

Hoofdstuk 1: Over het programma Profilab Expert 4.0



Eerst even iets over het programma. Als het programma geïnstalleerd en geopend is krijgen we dit te zien. Dit is het werkblad waar de schakeling op gemaakt wordt.

In de bovenste balk zien we horizontaal: **Datei, Bearbeiten, Frontplatte, Konfiguration, Optionen, Fenster, Registrierung** en een vraagteken. Als je met de muis op de icoontjes gaat staan zie je vanzelf wat je er mee kunt.

In **Datei**: nieuw, openen, opslaan, opslaan als, voorbeelden, macro importeren, macro opslaan, afdrukken en compileren.

Verticaal links van boven naar beneden: **Standaard, Verbindungen, Zoom, Start, Stop, Leitungszustand anzeigen, Bauteilzustand anzeigen, Frontplatte anzeigen, en Bauteilhinweise anzeigen.**

Met **Leitungszustand** kun je laten zien (simuleren) welke verbindinglijnen er hoog of laag worden als de schakeling draait. Zo kun je zien hoe de schakeling werkt. En het is ook makkelijk om fouten op te sporen in de schakeling als deze niet goed werkt.

Met **Verbindungen** kun je de componenten verbinden, bijv. een schakelaar met een led of poorten onderling, enz. enz.

Als je op het icoon naast standaard drukt kun je kiezen uit: **Frontplattenelementen, Logikbauteile, Analoges, Diverses, Hardware, en Makros.**

Frontplattenelementen is weer onderverdeeld in: **Anzeigen, Bedienung analog, Bedienung digital, Schreiber en Uhren.** Ik ga ze niet allemaal op noemen, dat zijn er iets teveel. Door op de verschillende iconen te gaan staan wijst alles voor zich.

Bij veel modules kun je onder eigenschappen nog van alles aanpassen, zoals waardes, kleur, vorm, spanning, enz. enz. Maar daar over later meer.

Wat zit er onder meer allemaal in het programma?

Digitaal: allerlei ledjes, lampen, displays, meters digitaal en analoog, klokken, timers, potmeters digitaal en analoog, ingaven modules digitaal en analoog, schakelaars digitaal en analoog, schrijvers, scoopfunctie, analyzer, plotter, AD en DA converters, decoders, flipflops, digitale poorten, multiplexers, RAM en ROM modules, registers, schuifregisters, tijdgevers, tellers, enz. enz.

Analoog: berekeningen, multiplexers, generatoren, regelaars, vergelijkers, versterkers, relais, en allerlei andere zaken.

Diversen: plus, min en massa, test, Server module, Hotkeys, Strings, Frequentieteller, DLL import, Soundmodules, enz. enz.

Hardware: algemeen pc (toegang tot verschillende poorten), MIDI, Modbus RTU, Modbus TCP, multimeter, OPC, relaiskaarten, RS232, Abacom, BMC, Conrad, ELV, Fischertechnik, H tronic, Hygrosens, Kolter, Labjack, Meilhaus, Velleman en nog veel meer.

Dit is een kleine opsomming wat er zo'n beetje in het programma zit en er zijn regelmatig updates ter beschikking. Die kunnen gratis gedownload worden bij Abacom.

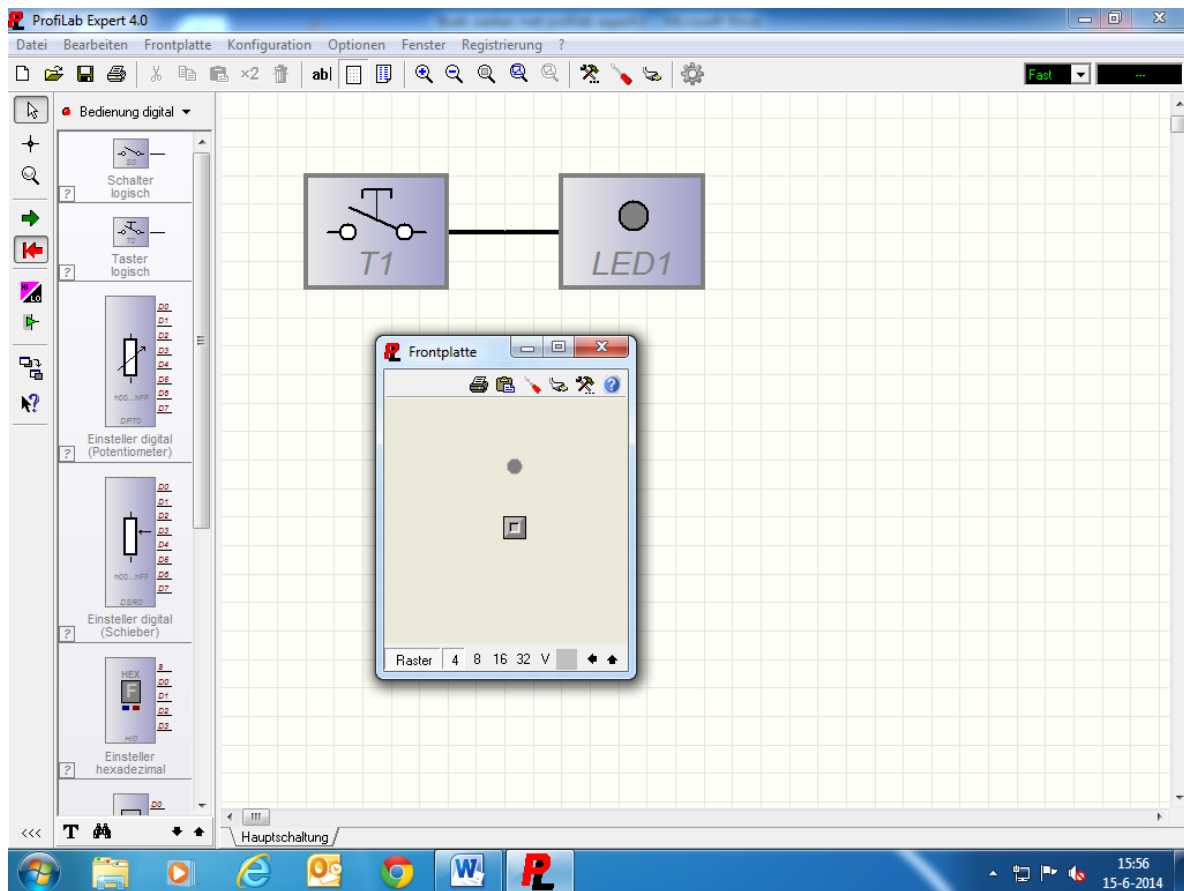
De opsommingen onder hardware zijn bijna allemaal fabrikanten die interfaces leveren voor Profilab. Behalve algemeen pc en RS232, daar kun je direct bij. Voor sommige is er wel een interface kaart nodig, maar bij sommige ook niet. Je kunt bijv. direct bij je parallelle poort, compoort en de internet poort (via de TCP module). Als je pc nog over serieel en parallel beschikt, kun je daar leuke dingen mee doen.

Met de multimeter module kun je direct een multimeter aan de pc hangen via de seriële poort, de multimeter moet daar wel geschikt voor zijn natuurlijk. Als je een geschikte meter hebt kun je daar van alles mee doen.

Voorbeeld: Je kunt de spanning meten en via een schrijver een paar dagen het lichtnet meten om te zien of er veel spanningsdippen op het net zijn. Dat kan ook met de stroom natuurlijk; je kunt er ook een vermogensmeter mee maken $P = U \times I$ en zo zijn er nog tal van voorbeelden op te noemen.

Zo, genoeg over het programma zelf. We gaan nu de eerste schakeling maken met Profilab.

Hoofdstuk 2: Drukknoppen en leds



Voor de eerste schakeling gebruiken we een digitale schakelaar en een led. Bij de schakelaar kun je kiezen tussen een drukschakelaar of een wipschakelaar. Die schakelaars zijn weer onder te verdelen in diverse verschillende types. Door nu op de knop te klikken naast standaard knop met de pijl en daarna op **frontplattenelemente** gaat staan, dan kun je **anzeigen** kiezen. Daar staan diverse componenten en modules voor op de frontplaat we kiezen de eerste, een gewone led.

Vervolgens doe we dit nog eens en nu kiezen we **bedienung digital**. Daar pakken we de tweede schakelaar **taster**, dat is een normale druk toets. Om de componenten op het werkblad te zetten hoef je ze alleen maar aan te klikken met de linker muisknop. Door nu op het component te gaan staan en links te klikken kun je het component verslepen. Als je rechts klikt op de componenten krijg je een venster te zien met: **eigenschaften, knippen, plakken, kopiëren, naar voren zetten, naar achter zetten, en draaien**. Daar moet je maar een beetje mee spelen om te zien wat je daar mee kunt doen.

We gaan nu de twee componenten aan elkaar zetten. Als de lijntjes aan elkaar zitten is het goed. Als we nu naar **Frontplatte anzeigen** gaan, dan zie je de componenten op de frontplaat staan. Deze componenten kun je nu verslepen naar een plaats die voor jou goed is. Zie voorbeeld hierboven.

Onder aan de frontplaat zie je **raster** staan, met daarnaast 4,8,16,32, en V. Met V kun je het raster zichtbaar maken en met 4 t/m 32 bepaal je het rasterformaat. Daarnaast staat een gekleurd vierkantje, daar kun je de kleur van de frontplaat mee veranderen. Met de pijlen daarnaast kun je de componenten opschuiven als die geselecteerd zijn. Daarnaast staan nog wat tekens voor het automatisch uitlijnen van de componenten. Zet de componenten op een plaats waar jij ze wilt hebben, selecteer ze met je muis door links te klikken en er een kadertje omheen te zetten. Als er een kader omheen staat kun je op één van de tekens drukken om de componenten automatisch uit te lijnen.

Als je nu rechts op een component drukt (bijv. de schakelaar) en je kiest eigenschappen, dan kun je van alles aan de schakelaar aanpassen. Je kunt dan bijv. een tekst ingeven, de kleur veranderen, of er een draaischakelaar van maken. Je kunt ook een bitmap koppelen aan de schakelaar voor een bepaald plaatje en nog veel meer.

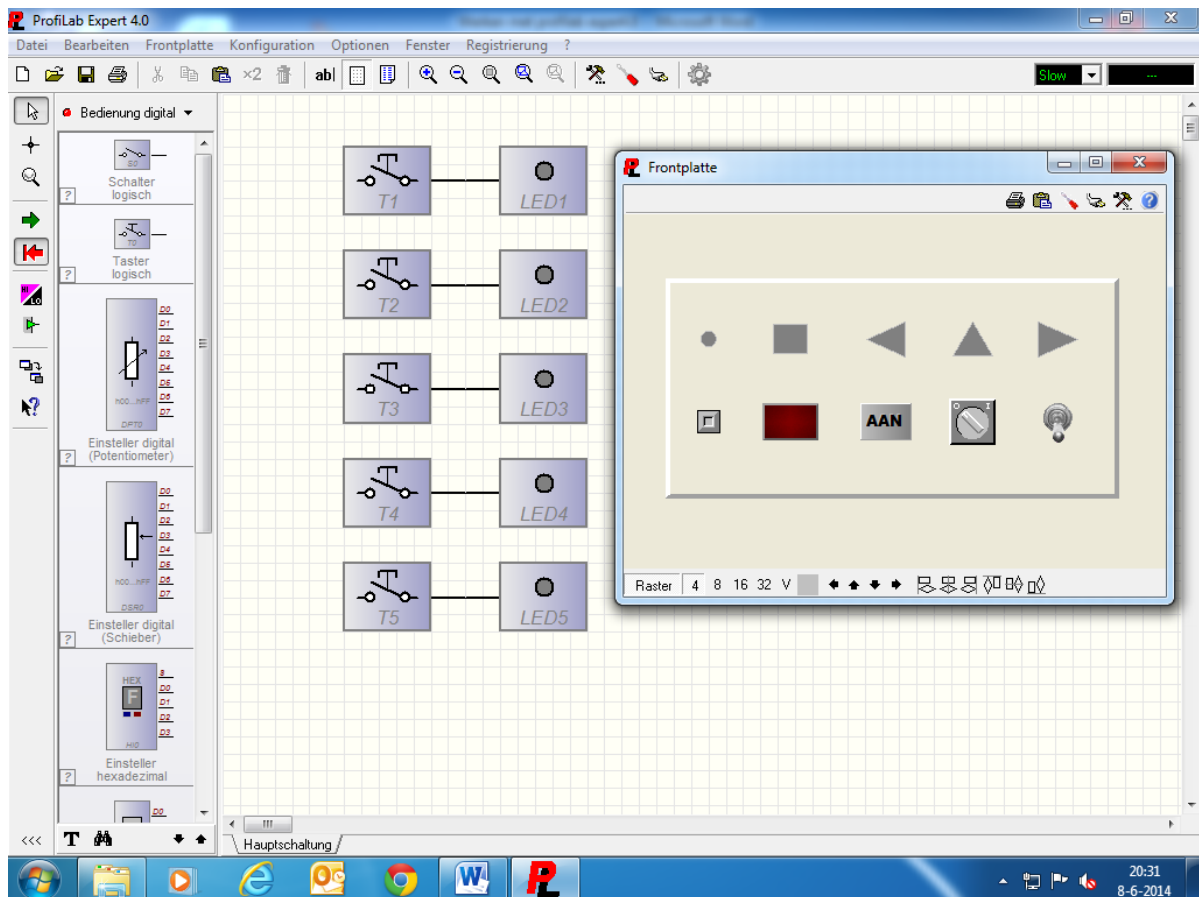
Bekijk de mogelijkheden maar eens rustig, en probeer maar eens wat uit. Dit geldt ook voor de led: zodra het component op de frontplaat staat en je kunt de eigenschappen zichtbaar maken, kun je van alles veranderen.

Nu gaan we de simulatie starten. Als we nu op de **startknop** drukken (**knop onder zoom**), kun je het programma starten. Als je alles goed gedaan hebt verschijnt de frontplaat nu op het scherm en kun je de knop bedienen. Als je nu op de drukknop drukt (met het pijltje op de knop gaan staan en op de linker muisknop drukken) zal de led gaan branden. Om uit de simulatie te gaan kun je de frontplaat afsluiten door rechts bovenin op het kruisje te klikken of door op de stopknop te drukken. Maar dat gaat eigenlijk alleen maar als je nog een kleine frontplaat hebt, dus houd voor gewoonte om de frontplaat af te sluiten.

We gaan nu de schakeling een paar keer kopiëren en een paar andere drukknoppen en andere leds maken.

We gaan zowel de kleur als de vorm aanpassen. We kunnen de schakeling gewoon op het werkblad selecteren en een aantal maal kopiëren en invoegen. Let er wel op als je kopieert en plakt, dan staan alle componenten op één plaats op de frontplaat. Het lijkt er dan dus op dat er maar één component staat, maar als je er op gaat staan kun je de componenten verslepen op de frontplaat. Je kunt natuurlijk ook gewoon een paar drukknoppen en leds naar het werkblad slepen. Ook gaan we een extra raam op de frontplaat maken. Met dat extra raam komt de frontplaat er mooier uit te zien en je kunt op deze manier groepen maken die qua bediening bij elkaar horen.

De afbeelding komt er dan uit te zien zoals op Blz. 11 staat.



We hebben nu vijf drukknoppen en vijf leds, allemaal in een andere vorm en een andere kleur. Voor de leds heb ik de kleuren (van links naar rechts), groen, groen, geel, blauw, en rood gekozen. Voor de drukknoppen één standaard-, één met verlichting-, één met tekst-, één draai-, en één steelschakelaar. Je kunt de kleur en de vorm weer veranderen onder **eigenschappen** van de componenten, net als in het eerste voorbeeld besproken.

Je kunt nu ook het extra raam zien met de bedienelementen erin. Hoe maken we zo'n raam?

Als je met de rechter muisknop op de frontplaat klikt, krijg je een scherm te zien met verschillende functies. Onder eigenschappen kun je een lijst aanvinken met diverse functies voor de frontplaat. De functies die aangevinkt zijn komen rechts boven in de frontplaat te staan. In hetzelfde scherm staan ook nog wat andere functies, onder andere ook de functie om een raam te maken. Als je de functie **neuer rahmen** kiest krijg je een verdiept vakje te zien. Ook hier kun je onder **eigenschappen** het raam veranderen; je kunt verschillende ramen kiezen en ook of het raam verdiept of verhoogd moet zijn. In het voorbeeld hebben we voor verhoogd gekozen.

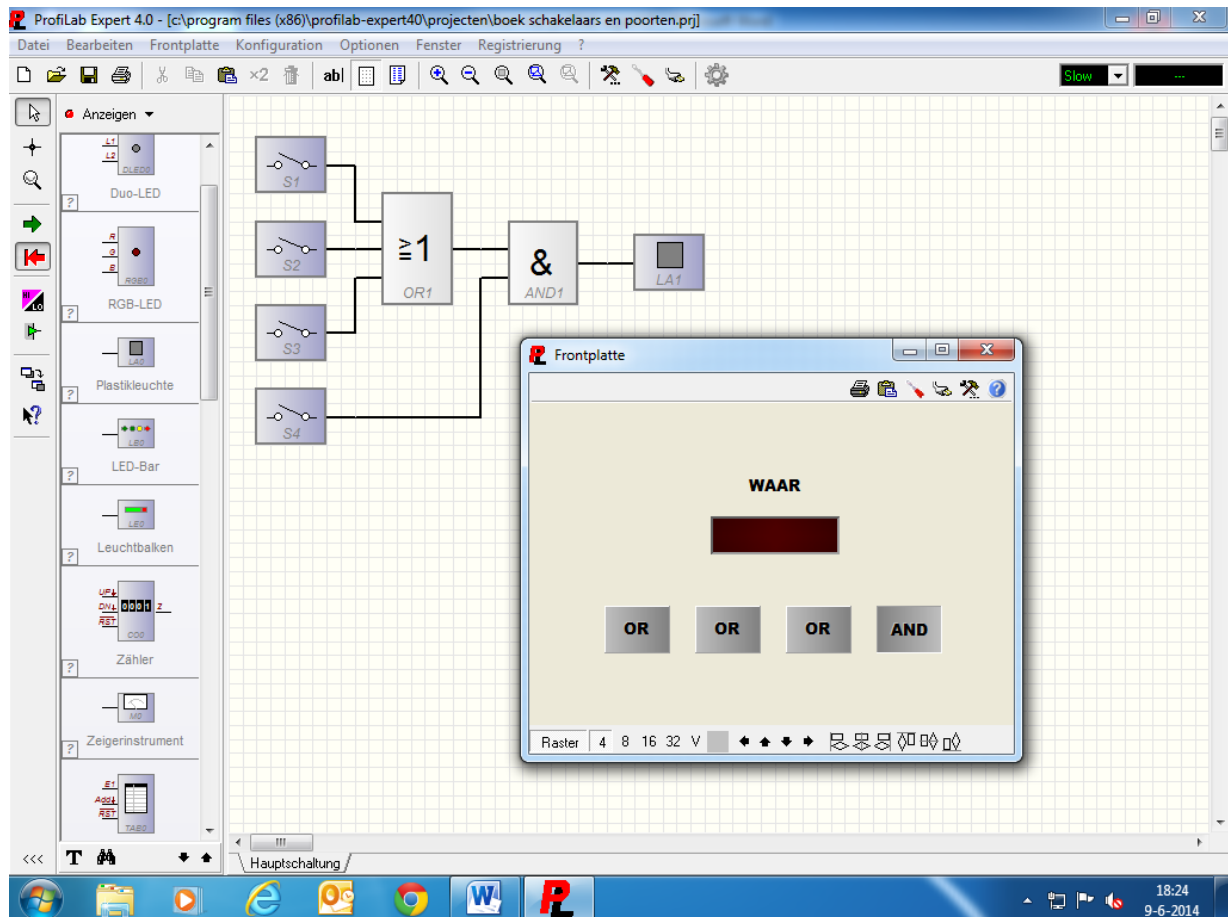
Als je nu nogmaals op het raam klikt (met de rechter muisknop) kun je **rahmen & hintergrund** kiezen. Daar kiezen we nogmaals rand verhoogd en de hoogte zetten we op drie en druk op OK.

Nu zien we dat het raam een verhoogde rand heeft. Wat we waarschijnlijk ook zien, is dat we de componenten niet meer zien. Door weer naar het venster te gaan en **nach hinten setzen** te kiezen, wordt het venster naar de achtergrond verplaatst en de componenten zijn weer zichtbaar.

Als je nogmaals op het raam klikt krijg je kleine vierkante blokjes te zien (zowel horizontaal als verticaal). Als je op die blokjes gaat staan met de muis en op de linker muisknop klikt kun je het raam groter of kleiner maken.

Als we nu weer op **start** drukken kunnen we de schakeling weer simuleren; het zijn drukknoppen dus om de leds te laten branden moet de knop vastgehouden worden. Met een normale schakelaar hoeft dat niet, die kun je gewoon aan en uit zetten. Maar daarover later meer.

2.1: Schakelaars en poorten



We gaan nu een schakeling maken met schakelaars en poorten.

We gaan weer naar de componentenlijst en kiezen **frontplattenelementen** en dan **bedienung digital**. We nemen nu de bovenste schakelaar en slepen er vier van naar het werkblad. We gaan weer terug naar **frontplattenelementen** en dan naar **logikbauteile** en kiezen daar **gatter**. We krijgen nu digitale poorten te zien en daar pakken we één OR poort en één AND poort.

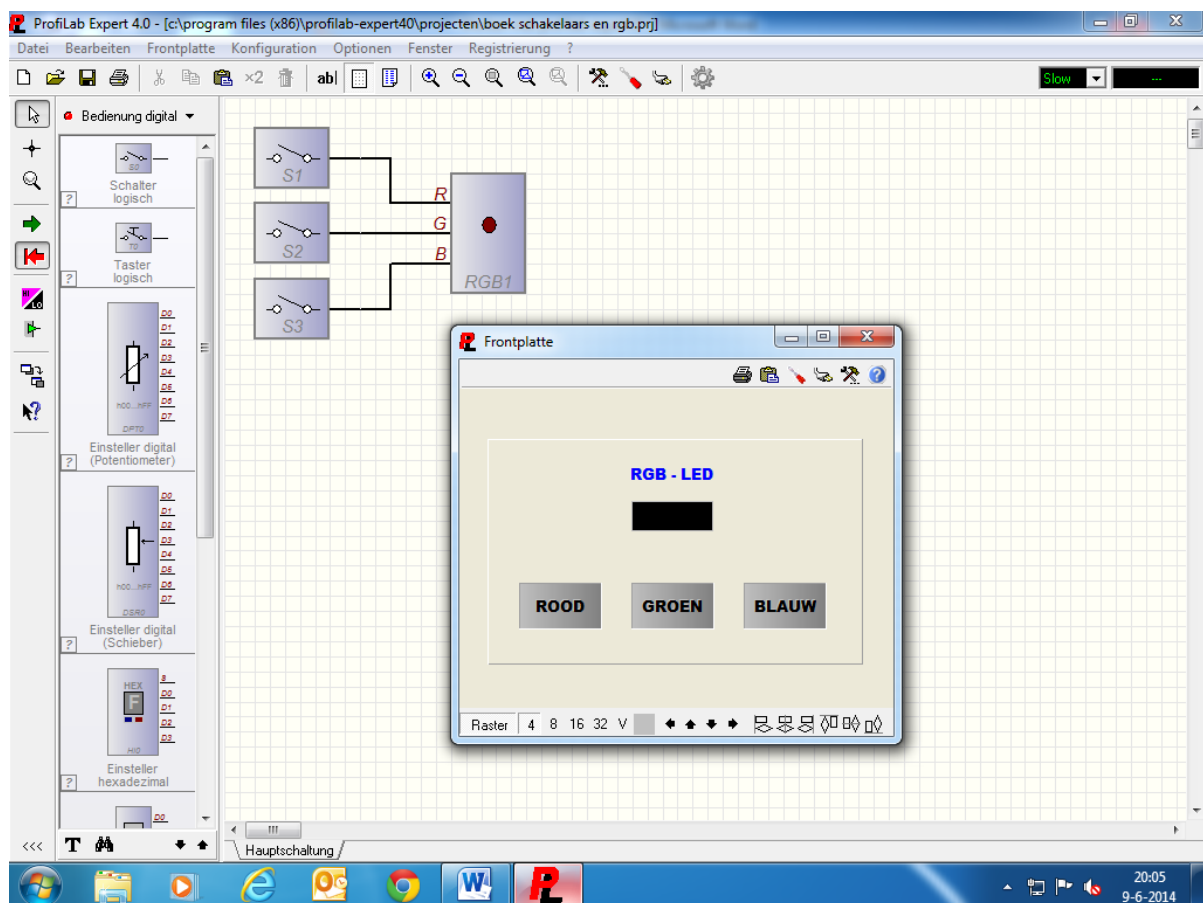
We slepen deze weer naar het werkblad. We gaan weer terug naar de componentenlijst, kiezen **anzeigen** en slepen een lampje (vierde van boven) naar het werkblad. Onder eigenschappen van de OR poort kiezen we nu een poort met drie ingangen. We hebben nu alle componenten en gaan ze samen voegen zoals op het werkblad te zien is.

Dat kun met het knopje **verbindungen**, dat is het knopje boven het vergrootglas in de linkerbalk. Als we nu naar de frontplaat gaan, kunnen we de schakelaars en de lamp netjes uitlijnen. Onder eigenschappen van de schakelaars kiezen we nu

schaltfläche en zetten bij drie schakelaars OR neer en bij één schakelaar AND. We maken de schakelaars iets groter, zodat de tekst goed zichtbaar is. Als we nu op de frontplaat klikken en kiezen in de lijst *neuer tekst*, dan kunnen we een tekst op de frontplaat zetten. Je kunt kiezen uit vele tekstsoorten en -kleuren. We zetten de tekst **waar** neer. Als we nu de simulatie starten en we drukken op de OR schakelaars, zal er niets gebeuren. Maar als de AND schakelaar ingedrukt wordt, zal de lamp gaan branden onder de tekst **waar**. De schakelaars zijn van het type houd: als je één keer op de schakelaar drukt is hij aan, druk je er nog één keer op dan is de schakelaar uit.

Onder eigenschappen van de schakelaar kun je ook nog kiezen of de schakelaar aan of uit is als je de simulatie opstart. Dat kan je doen onder **default**, rechts bovenin het venster.

2.2: Schakelaars en RGB



We gaan nu een schakeling maken met een RGB led. Door middel van de drie schakelaars kun je de kleur bepalen van de RGB led. We gaan weer naar de componentenlijst en kiezen **frontplattenelementen** en vervolgens **bedienung digital**. We slepen nu drie schakelaars naar het werkblad.

We gaan weer terug naar **frontplattenelementen** en dan naar **anzeigen** en daar pakken we de RGB led; deze slepen we weer naar het werkblad. We verbinden de componenten weer met elkaar, zoals je kunt zien in het voorbeeld. We gaan nu weer naar de frontplaat en lijnen alles netjes uit. Onder eigenschappen van de schakelaars kiezen we weer **schaltfläche** en zetten bij schakelaar_1 rood, bij schakelaar_2 groen, en bij schakelaar_3 blauw. De schakelaars maken we weer wat groter zodat de tekst weer goed zichtbaar wordt. De RGB led maken we ook wat groter.

We klikken weer rechts met de muis op de frontplaat en kiezen weer **neuer tekst**. We zetten nu de tekst RGB – LED neer en maken de tekst blauw. We gaan er ook weer een raam omheen zetten, we kiezen dus weer onder eigenschappen **neuer rahmen**. Onder eigenschappen van het raam kiezen we nu **linie** en dan **erhöhung**. We gaan het raam niet verder verhogen zoals in het voorbeeld bij hoofdstuk twee. We maken het raam weer wat groter en onder eigenschappen kiezen we weer **nach hinten setzen**. Als het goed is, is alles weer zichtbaar.

Als we nu de simulatie weer starten en we drukken op schakelaar_1, zal de RGB LED rood worden. Als we op schakelaar_2 drukken wordt de led groen en als we op schakelaar_3 drukken wordt de led blauw. Het is dus weer eenmaal op de schakelaar drukken is **aan**, en nogmaals drukken is **uit**.

We kunnen nu ook kleurencombinaties maken door verschillende schakelaars te combineren, probeer het maar eens uit.

In de volgende voorbeelden ga ik het niet meer hebben over hoe je een raam kunt maken of hoe je een tekst in een schakelaar zet of de wijze waarop je tekst op de frontplaat kunt zetten. Dat moet nu wel lukken denk ik. Als er nieuwe dingen zijn doe ik dat uiteraard wel natuurlijk.

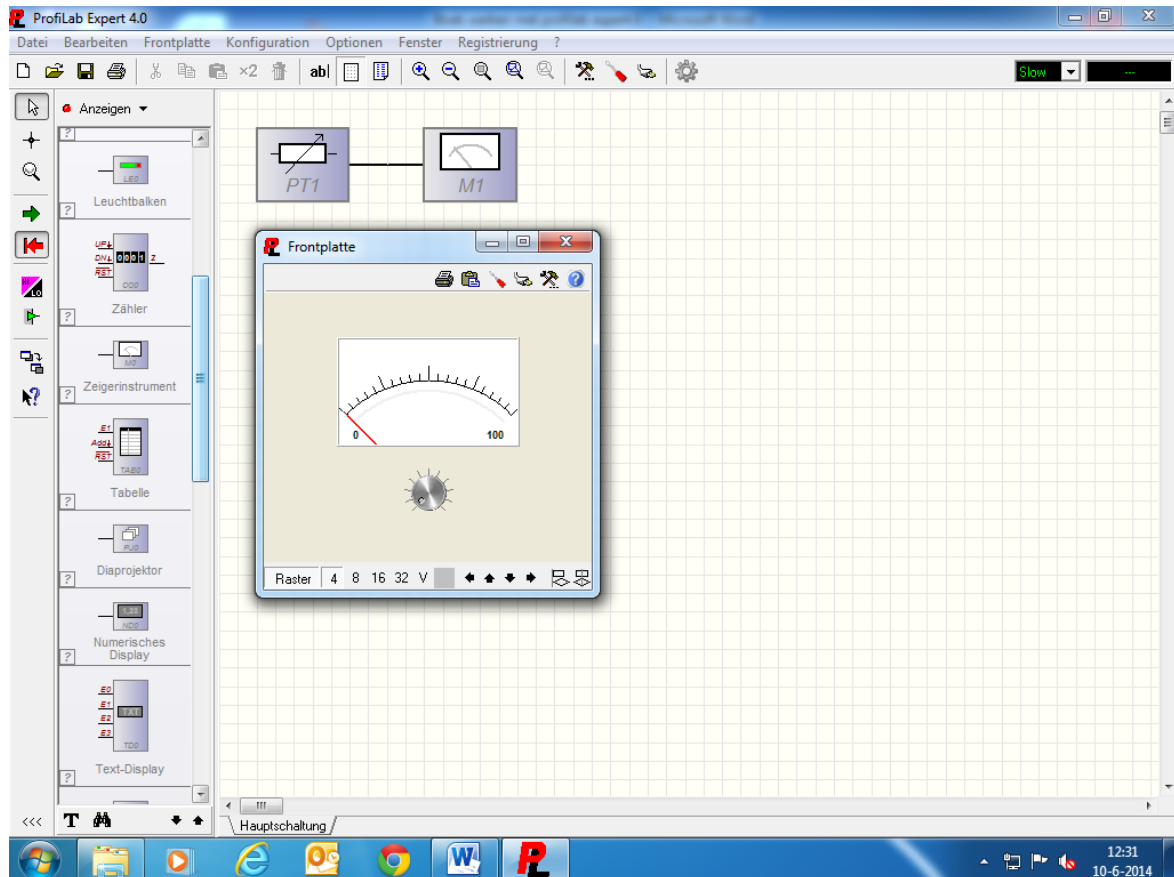
Hoofdstuk 3: Analoge elementen

We gaan nu iets doen met analoge elektronica, dat wordt nog steeds veel gebruikt. Denk maar eens aan sensoren; veel sensoren geven een analoog signaal af. Als je deze signalen binnen wilt halen met Profilab, heb je natuurlijk wel een interface nodig waar een analoge ingang op zit. Maar je kunt ook analoge componenten in je schakeling gebruiken onder Profilab. Dat kunnen potmeters zijn, aanwijsinstrumenten (zoals meters en displays), AD converters, DA converters, enz. enz.

We gaan weer naar de componenten- lijst, kiezen **frontplattenelementen** en dan **bedienung analog**. We slepen nu de bovenste potmeter naar het werkblad. We gaan weer naar **frontplattenelementen** en kiezen **anzeigen**. Daar pakken we een meet- instrument (zesde van boven) en slepen die naar het werkblad. We verbinden de componenten weer aan elkaar en gaan naar de frontplaat en we lijnen de componenten weer netjes uit. Als je nu op de potmeter klikt, kun je onder **eigenschaften** weer het één en ander instellen. Bij **bereich** vullen we in 0 tot 100, en **auflösung** zetten we ook op 100. Verder kun je bij **vorgabe** nog een waarde invullen als je niet op nul wilt beginnen, maar dat doen we nu niet. We gaan nu naar de meter en gaan onder **eigenschaften** de meter instellen. Bij de instelling **bereich** vullen we de waarde 100 in, als we de waarde van de potmeter en van de meter niet gelijk zouden houden krijgen we geen goede waarde op de meter. Bij **einheit** kun je aangeven waar de meter voor bedoeld is. Dat kan zijn: Ampère, Volt, Temperatuur, enz. enz. Wij vullen daar niets in, wat er staat halen we weg.

Bij **trägheid** kun je een waarde ingeven die er voor zorgt dat de meter traag reageert, net als bij een echte meter met een spoel. Als je de waarde op nul zet reageert de meter direct, dus zonder vertraging. Wij laten de waarde op 10 staan.

Op het voorbeeld hieronder zie je hoe het eruit komt te zien.



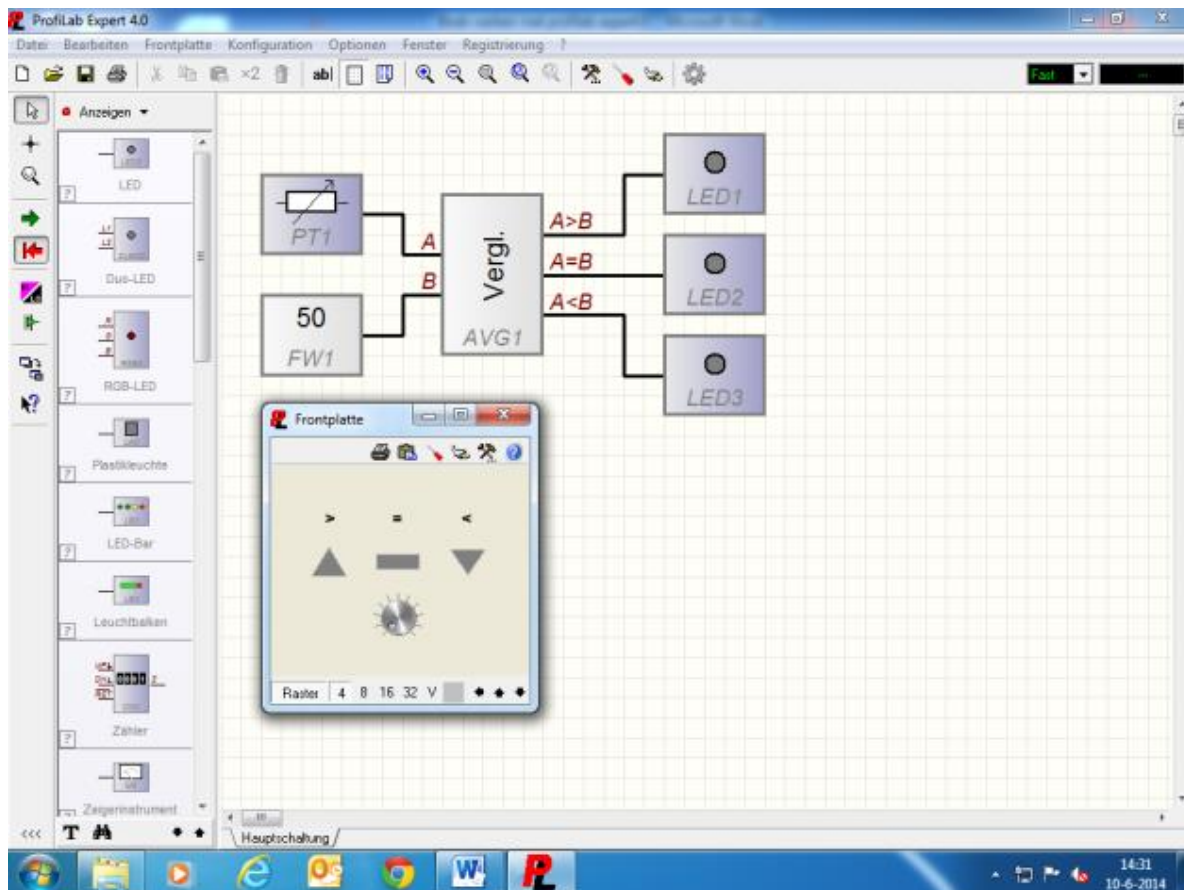
Als we alles ingesteld hebben gaan we weer naar de simulatie. Als je nu met de muis op de potmeter gaat staan en je klikt links, dan kun je als je de muis beweegt de potmeter verdraaien. De waarde wordt nu één op één doorgegeven aan de meter.

Je kan de potmeter ook nog anders bedienen: als je op de potmeter klikt komt er een pijltje bij te staan (is trouwens bij alle bedienelementen zo), als het pijltje erbij staat kun je de potmeter ook bedienen met **page up** en **page down** of **home** en **end**.

3.1: Analooq en digitaal combineren_1

We gaan nu wat analoge en digitale componenten combineren. Dat kan nodig zijn om alle functies optimaal te benutten met ProfiLab. Als je bijv. een vergelijker module pakt, die heeft twee analoge ingangen en drie digitale uitgangen. Of een AD converter met één analoge ingang en een acht bit digitale uitgang. Dit is dus allemaal met elkaar te combineren. We zullen er eens een vergelijker module bij pakken en daar een schakeling mee gaan maken.

Zie voorbeeld op blz. 16.



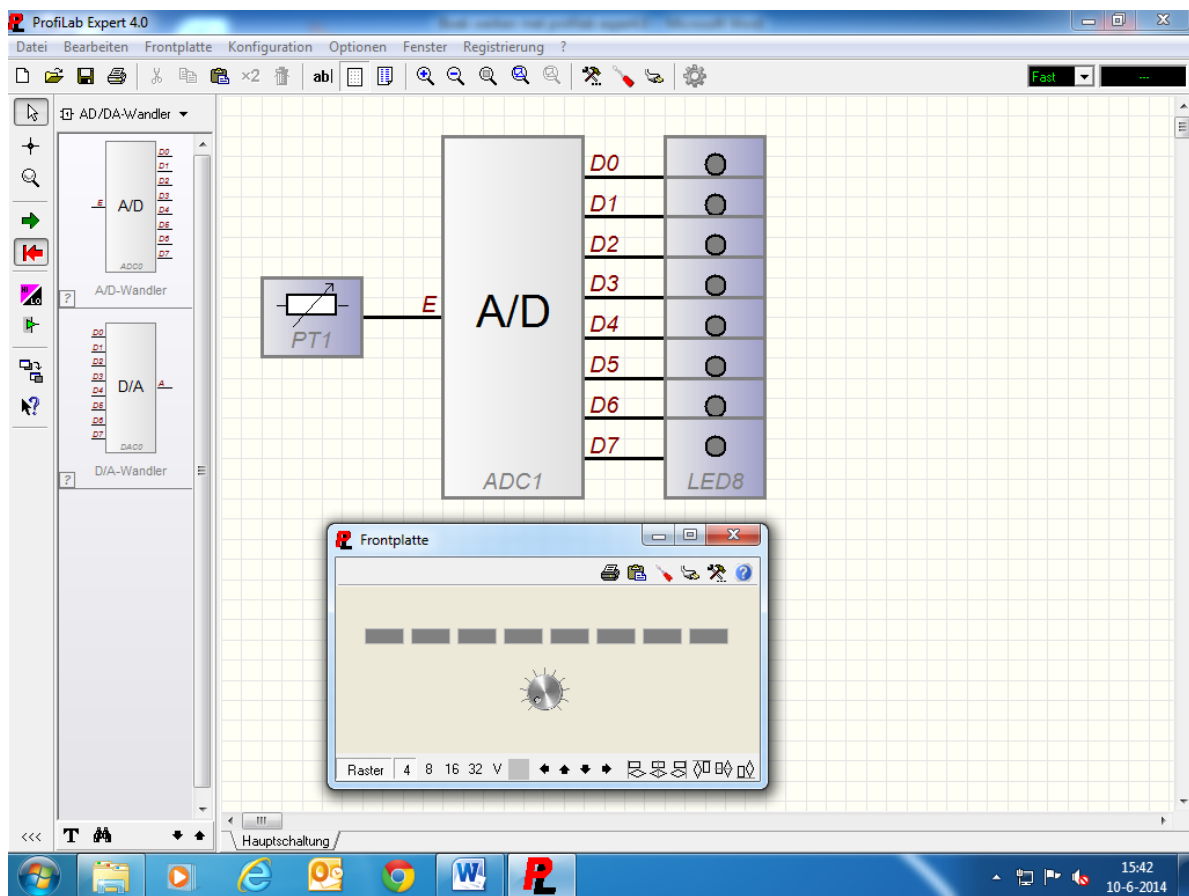
We gaan weer naar **frontplattenelementen** en dan naar **anzeigen**, en we slepen drie leds naar het werkblad. We gaan weer naar **frontplattenelementen** en dan naar **bedienung analog** en we pakken de bovenste potmeter en slepen ook deze naar het werkblad. We gaan nu naar **analoges** en dan naar **vergegleicher**, we slepen de tweede van boven naar het werkblad. We gaan weer naar **analoges** en dan naar **quellen**, daar pakken we de bovenste module en slepen die ook naar het werkblad. We hebben nu in het totaal zes componenten staan en daar gaan we dit mee maken.

We gaan de componenten net zo plaatsen als op het voorbeeld, dus de potmeter op ingang A, het vaste waarde module op ingang B en aan de digitale uitgangen komen de leds. Het vaste waarde module stellen we in op 50 onder **eigenschaften**.

Als alles weer aangesloten is gaan we weer naar de frontplaat om alles weer netjes uit te lijnen. We geven de potmeter weer een waarde van 0 tot 100, en het instelbereik zetten we ook weer op 100. Dus precies hetzelfde als bij Hoofdstuk 3: analoge elementen. Van twee leds heb ik weer driehoekjes gemaakt en van één led heb ik een rechthoek gemaakt. De middelste led is weer wat groter gemaakt en van de driehoek leds staat er één naar boven en één naar beneden. Ik heb daarboven in tekst gezet: > = <.

Als we nu de simulatie weer starten en we draaien aan de potmeter, dan zal je zien dat als de waarde > groter is dan 50, de meest linkse led gaat branden en als de waarde = gelijk is aan 50, de middelste led gaat branden en als de waarde < kleiner is dan 50, de meest rechtse led gaat branden.

3.2: Analog en digitaal combineren_2



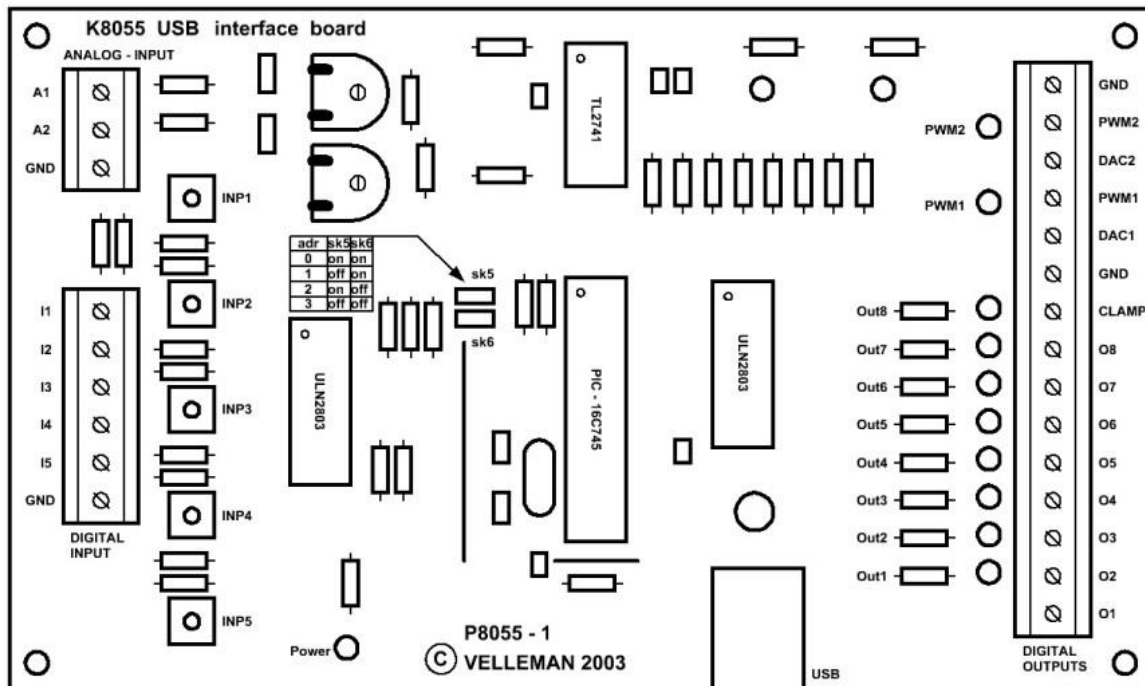
We gaan nu een schakeling opzetten met een potmeter en een AD converter. Ik ga nu niet meer zeggen waar we alles kunnen vinden, dat moet nu wel duidelijk zijn. Wel ga ik de componenten benoemen (dus wat we nodig hebben) en de instellingen van de componenten natuurlijk.

Wat hebben we nodig: eenmaal een potmeter, eenmaal een AD converter en achtmaal een led. Dit slepen we allemaal weer naar het werkblad. We verbinden alles zoals op het voorbeeld te zien is. De potmeter komt op ingang E van de AD converter en de leds komen aan de datalijnen D0 t/m D7, led 1 t/m led 8. De potmeter geven we nu een bereik van 0 tot 255, en bij **auflösung** geven we dezelfde waarde in, ook 255 dus. De AD converter stellen we in op 8 bits en het bereik ook van 0 tot 255. De AD converter is instelbaar tussen twee en zestien bits. Op de frontplaat gaan we alles weer netjes uitlijnen, er staat dit keer geen tekst bij. Maar als je dat wilt kan dat altijd natuurlijk.

Als we nu de simulatie opstarten en je aan de potmeter draait zal je zien dat de AD converter volloopt (van links naar rechts) en dat gebeurt dan met een binaire code.

Dus één is 00000001 twee is 00000010 drie is 00000011 vier is 00000100 vijf is 00000101 zes is 00000110 zeven is 00000111 enz. enz.

Hoofdstuk 4: De K8055



We hebben nu wat voorbeelden gemaakt, dus nu wordt het tijd om de K8055 erbij te pakken. Zoals al eerder gezegd is dit een interface van Velleman en wordt deze interface door Profilab ondersteund. Bij de kaart zit een driver, deze dient voor gebruik geïnstalleerd te worden op de pc.

Tot Windows XP werkt dat met de driver die erbij zit, maar vanaf Windows 7 en 8 moet er een driver van internet worden gehaald. Dit kan op de site van Velleman. Als de driver geïnstalleerd is kan de kaart aangesloten worden op een USB poort van de pc. Als alles goed verloopt zal de achtste led (dat is ook de achtste uitgang op de kaart) een aantal malen gaan knipperen en daarna uit gaan, de Power led op de kaart gaat ook branden. Op blz. 13 staat een afbeelding van de interface kaart de K8055.

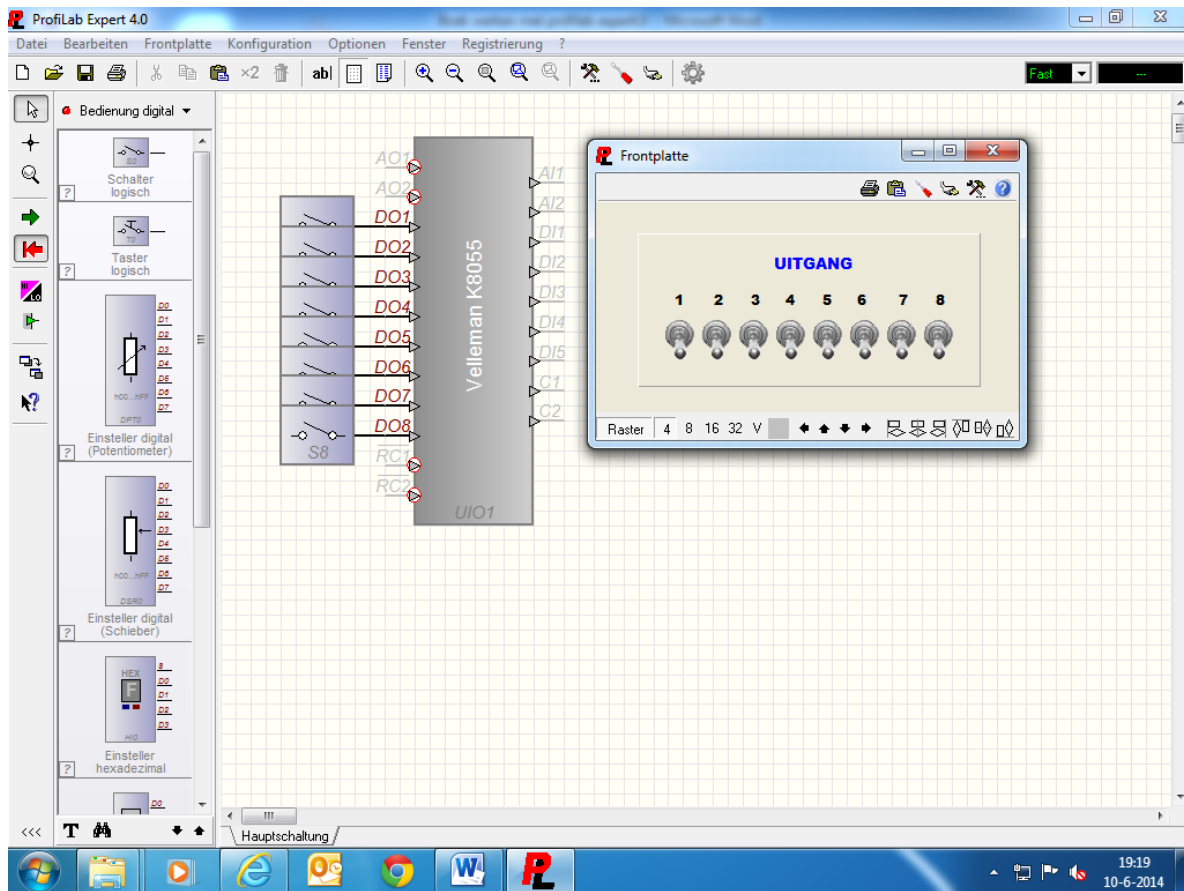
De interface kaart bevat: Twee maal een analoge ingang. Vijf maal een digitale ingang. Acht maal een digitale uitgang. Twee maal een analoge uitgang. En twee maal een PWM uitgang.

Daar zijn best leuke dingen mee te doen, zeker op experimenteel gebied.

Voor meer info over de K8055 kan je op internet een datasheet downloaden, daar staat alles in wat voor de interface van toepassing is.

Je moet als je met de PC werkt wel altijd bedenken dat Windows voorhang heeft op alles, dat betekent dus dat je met interrupts van Windows te maken hebt. Maar als het niet om een strakke timing gaat en om snelle signalen, dan is er prima mee te werken. Als snelle signalen en een strakke timing wel nodig zijn, kan dat op andere manieren opgelost worden. Maar daar over later meer, dat komt bij de microcontroller aan bod.

4.1: Digitale uitgangen

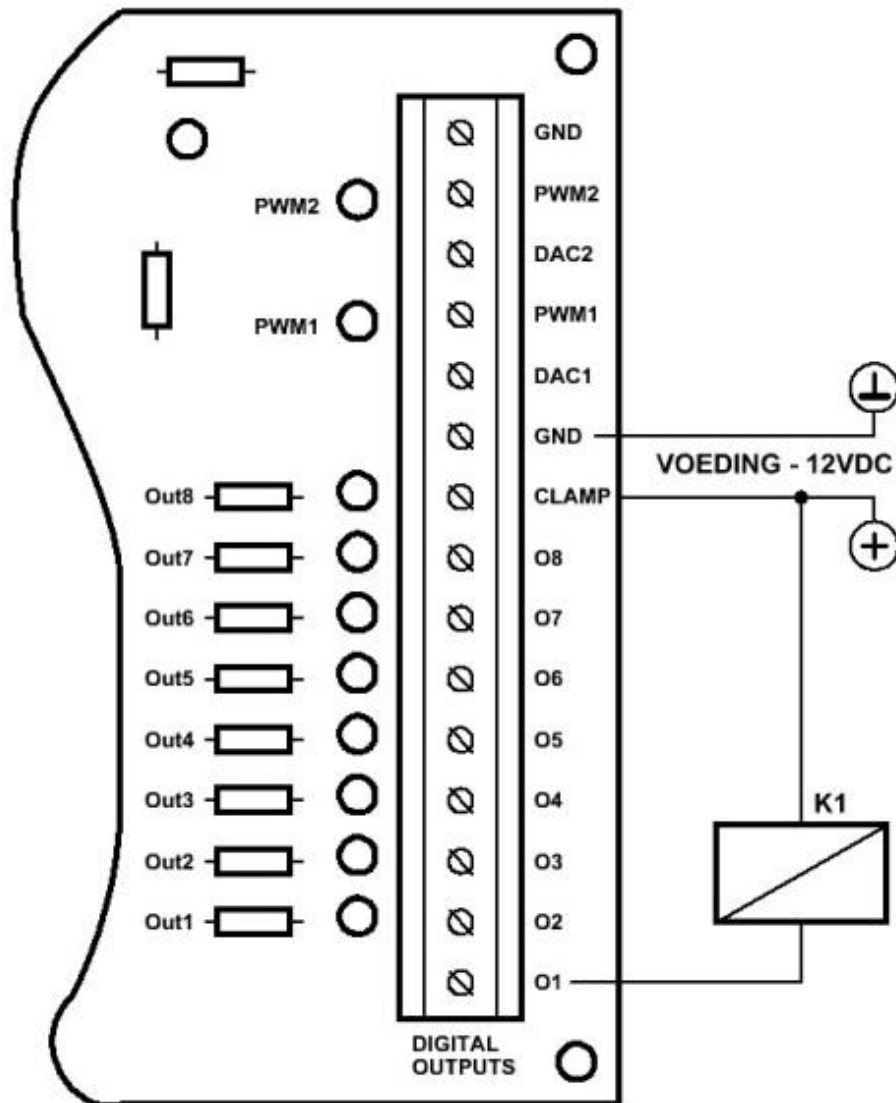


We gaan nu wat schakelingen maken met de K8055. De K8055 staat onder **hardware**, als we dat kiezen krijgen we een hele lijst te zien met allerlei fabrikanten die allemaal verschillende interfaces leveren. Helemaal onderaan in de lijst staat Velleman, als we daar naar toe gaan zien we bovenaan de K8055 staan. We slepen die naar het werkblad. Onder eigenschappen kan je een paar poort adressen selecteren (van 0 tot 3), wij laten het adres op nul staan. Met die adressen kan je meerdere kaarten aan de PC hangen, maar wij gaan in de voorbeelden maar één kaart gebruiken. Links onderin het venster (**onder eigenschappen**) staat nog een knop, als je die aan klikt krijg je een lijst te zien met de gegevens van de pinnen. Daarop kan je zien wat de in en uitgangen zijn, en wat je er op aan kan sluiten. Als je met de rechter muisknop op de module klikt kan je ook naar **hulp** of **hulp** **zu Velleman K8055 USB interface** gaan, daar kan je ook nog info vinden over de kaart. Dat is trouwens bij nog meer modules mogelijk.

We slepen ook nog acht schakelaars naar het werkblad, en sluiten deze aan op de digitale uitgangspinnen van de K8055. Schakelaar_1 komt op DO1, schakelaar_2 komt op DO2, enz. enz. Op de rest van de pinnen (aan de linker kant van de module) klikken we een keer met de muis. Daar zie je nu een rondje om de pinnen verschijnen, dat betekent dat die pin geïnverteerd is en dus laag is. Als je dit niet zou doen, dan branden de ledjes bij de analoge uitgang en dat is niet nodig.

Er is ook weer een raam gemaakt, en er staat wat tekst op de frontplaat. De schakelaars zijn aangeduid met één t/m acht, en de tekst uitgang staat erbij en deze heeft weer een kleurtje.

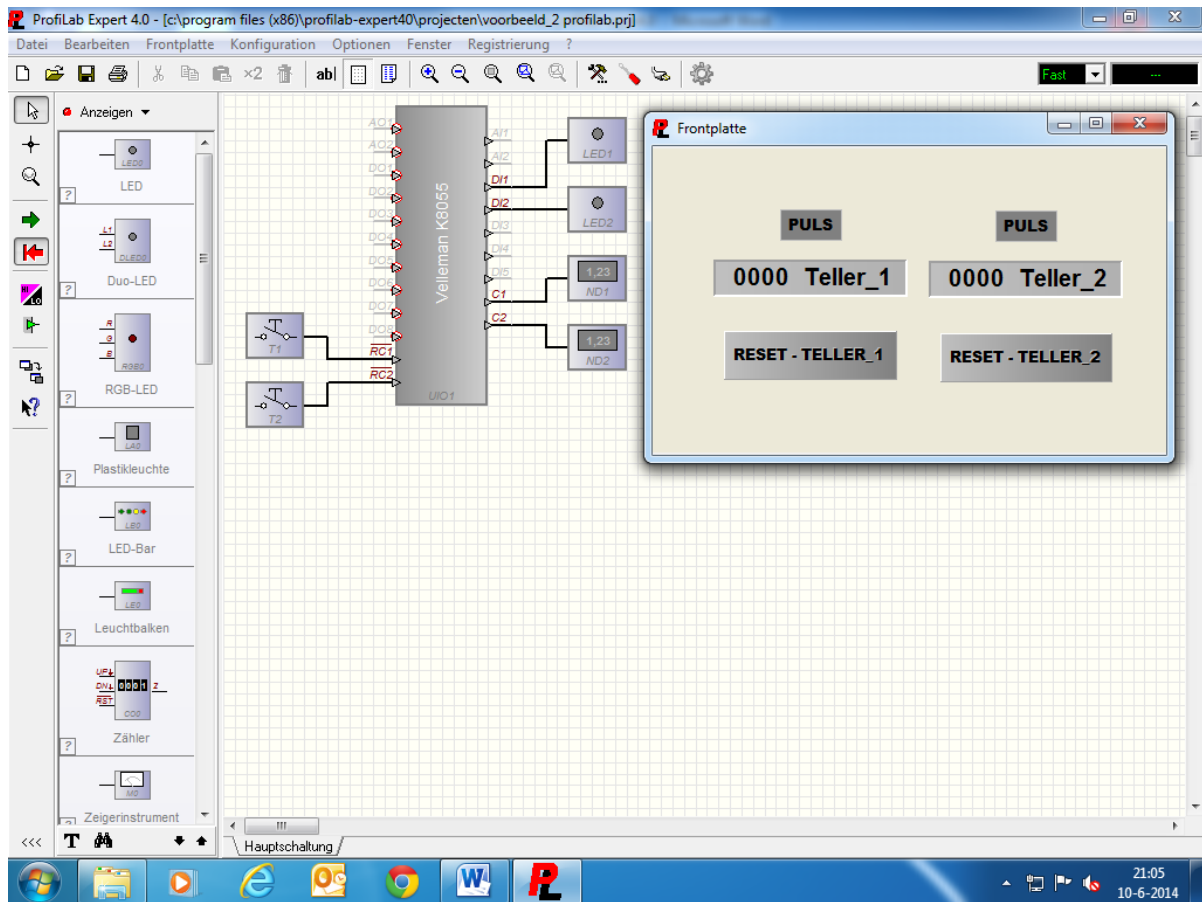
Als we nu de simulatie weer starten, kan je via de schakelaars de uitgangen één tot en met acht op de K8055 bedienen. Aan de uitgangen kan je bijvoorbeeld lampjes hangen, of je bediend er relais mee.



Aan de uitgangskant op de K8055 (**waar de printkroonsteen zit**) kan je een externe spanning aansluiten, de plus komt dan op de clamp aansluiting en de 0VDC komt aan de GND aansluiting. Zie voorbeeld hierboven.

Het is niet nodig om een vrijloop diode over de relaispoel te zetten, die zitten al in het driver IC de ULN2803. De belasting (in dit geval het relais) wordt naar de GND getrokken, de K8055 schakelt dus in de min.

4.2: Teller



We gaan weer een nieuwe schakeling opzetten rond de K8055, en we gaan nu de teller functie van de kaart gebruiken. Er zitten twee teller ingangen op de kaart RC1 en RC2, die worden geactiveerd door de drukknoppen INP1 en INP2 op de kaart of door extern een schakelaar op de printkroonsteen aan te sluiten aan de ingang kant van de kaart.

We gaan weer wat componenten naar het werkblad slepen. We pakken tweemaal een drukknop, tweemaal een led, en tweemaal een display. Het display staat onder **anzeigen** net als de led, het display staat elfde van boven. We gaan het geheel weer samenvoegen, de twee drukknoppen komen aan RC1 en RC2, de leds komen aan DI1 en DI2, en de displays komen aan C1 en C2.

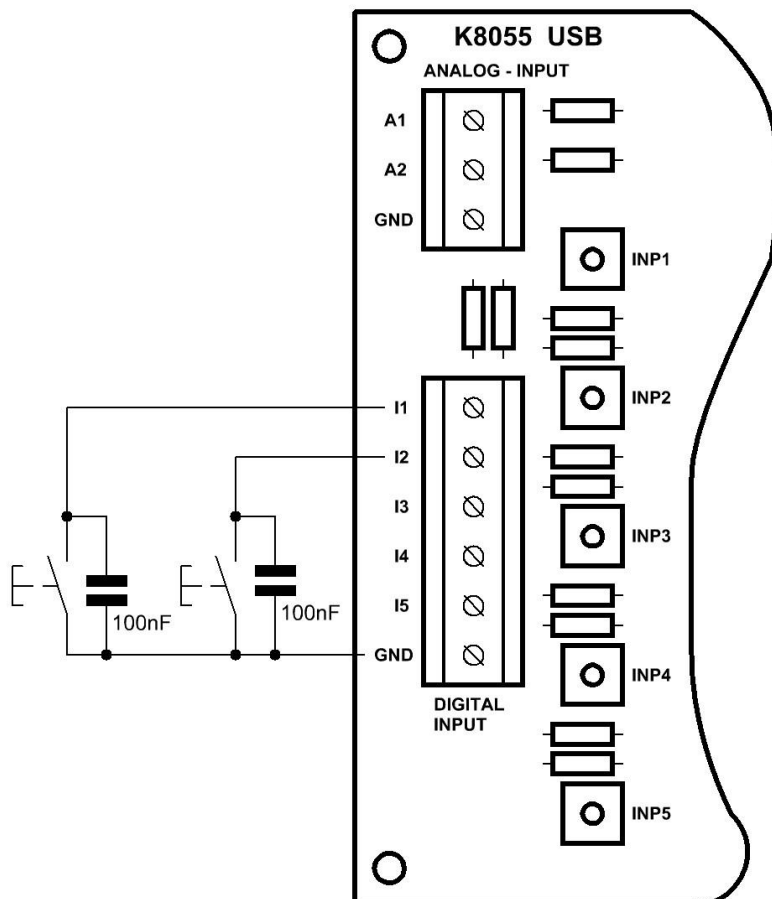
De ongebruikte uitgangen van de K8055 module zetten we weer als geïnverteerd door op de uitgangen te klikken, er komt weer een rondje om de pinnen te staan.

Als alles netjes aan elkaar zit gaan we weer naar de frontplaat. In de leds gaan we deze keer een tekst zetten, die tekst wordt **puls**. We gaan nu het display ND1 en ND2 instellen, dit doen we weer onder eigenschappen.

Als we het venster geopend hebben zie je **eenheit** staan, daar zetten we teller_1 neer. In dat zelfde venster zien we achter **fontmat - fest** een drukknop staan, als we daar links op klikken krijg je een nieuw venster te zien.

Daar klikken we **komma fest** aan, en zetten **vor komma** op vier en **nach komma** op nul. Nu hebben we een display met een vier digit uitlezing. Deze instelling doen we ook bij ND2. De drukknoppen geven we de tekst reset – teller_1 en reset – teller_2 mee.

Nu we alles ingesteld hebben kunnen we alles netjes uit gaan lijnen op de frontplaat, als dat gedaan is gaan we weer naar de simulatie.

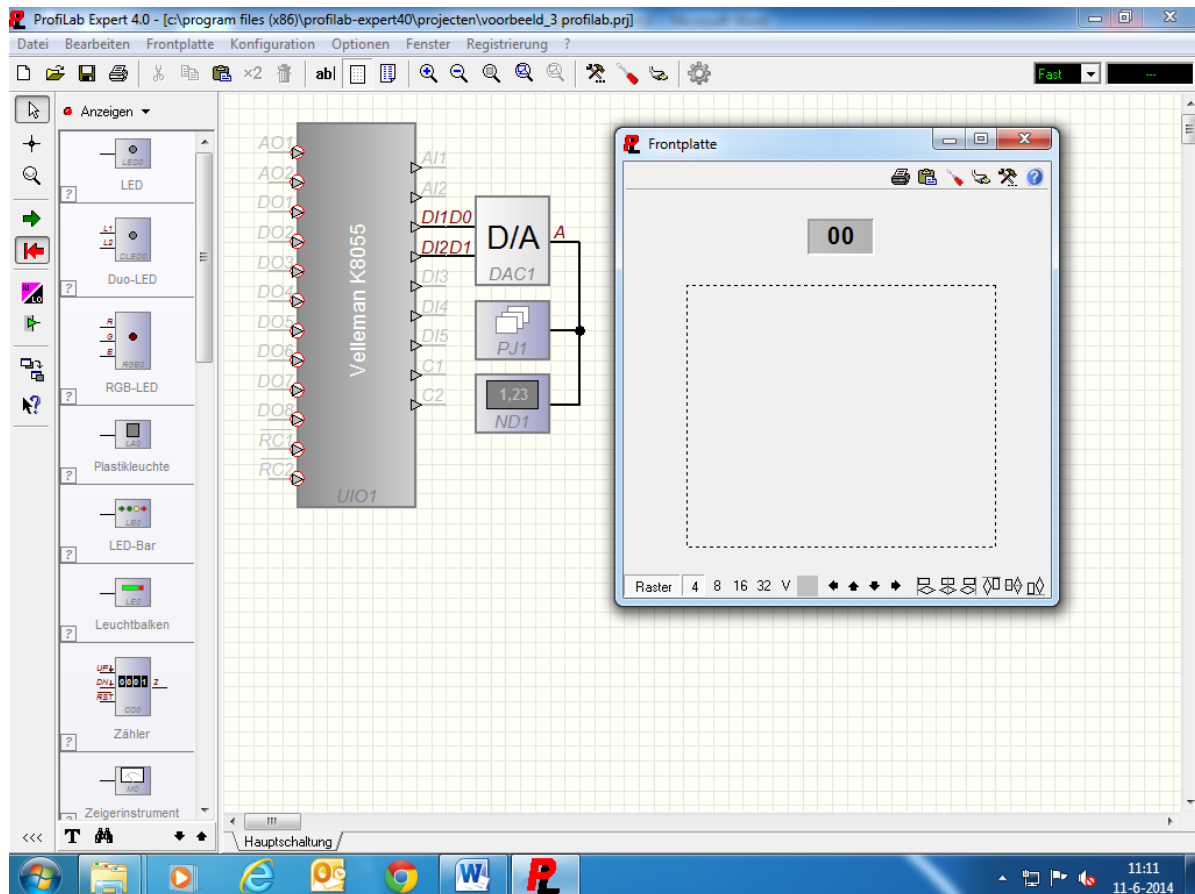


Als je nu op de knop INP1 drukt (drukknop op de K8055) zal teller_1 op de frontplaat ophogen, en de led (met de tekst puls erin) zal oplichten. Om de teller te resetten moet er op reset – teller_1 gedrukt worden op de frontplaat. Het zelfde verhaal geldt ook voor teller twee, deze teller kan geactiveerd worden door de drukknop INP2 op de K8055. Zo kan je dus op eenvoudige wijze een teller maken die ook nog eens vlot reageert. Met de inkomende data kan je natuurlijk van alles en nog wat doen, maar daar kan mee geëxperimenteerd worden natuurlijk.

Hier nog even een schema hoe je de externe ingangen van de K8055 aan kan sluiten via de printkroonsteen. Zoals te zien is op het schema staan er ook een paar 100nF condensatoren over de drukknoppen, dit is om contact dender tegen te gaan voor foutloos tellen.

Als de kabels wat langer worden (tussen de drukknop en de ingang) kan er ook nog een optocoupler tussen de drukknop en de ingang gezet worden. De optocoupler moet dan wel zo dicht mogelijk bij de K8055 komen te staan. De collector komt dan aan ingang_1 en de emitter komt aan de GND. De led in de optocoupler kan dan aangestuurd worden met bijv. een extern contact. Dat kan een contact zijn van diverse sensoren. Dat kan bijv. een inductieve sensor zijn of een lichtsluis, of anders natuurlijk.

4.3: Projector



We gaan weer een nieuwe schakeling opzetten rond de K8055, we gaan nu een projector maken. Je kan de projector met de K8055 bedienen, we gebruiken ingang_1 (DI1) en ingang_2 (DI2) hiervoor.

We slepen een DA converter, een projector, en een display naar het werkblad. De DA converter staat onder **logik bauteile** en dan **AD/DA wandler**. De projector en het display staan onder **anzeigen**. We zien bij het voorbeeld dat de projector en het display gedraaid zijn, als je rechts klikt op de projector dan zie je in het venster **drehen** staan als je daar op klikt draait het component.

We gaan alles weer verbinden zoals het op het voorbeeld te zien is. De converter komt op DI1 en DI2 van de K8055, en op de uitgang van de DA converter komt de projector en het display.

We gaan eerst de converter instellen, we zetten de converter onder **auflösung** op twee bits en onder **bereich** min/max zetten we de minimum waarde op nul en de maximum waarde op drie. Nu stellen we de projector in, hiervoor gaan we naar de frontplaat. We klikken rechts op de projector en gaan naar eigenschappen daar klikken we links op **bild hinzufügen**, nu kan je een GIF, JPG, JPEG, BMP, ICO, EMF, of een WMF bestand laden. Als je een bestand geladen hebt, kan je het boven genoemde herhalen voor meer plaatjes. Wij gaan nu drie plaatjes laden, omdat we in het voorbeeld maar een twee bit AD converter gebruiken. We gaan nu het display instellen, onder eigenschappen komen we weer bij het knopje dat achter **format / fest** staat. Als we daar op klikken krijgen we het venster te zien waar we het display in kunnen stellen. We selecteren **koma fest** en zetten **vorkomma** op twee, en **nachkomma** op nul.

We hebben nu een twee digit display, dat is voor dit voorbeeld ruim voldoende. We gaan op de frontplaat de projector en het display weer onder elkaar zetten, als we dat gedaan hebben starten we de simulatie weer. Je zal zien dat alleen het display zichtbaar is, maar zodra je op de knop INP1 drukt zal het eerste plaatje op de frontplaat verschijnen. Druk je nu op INP2 dan verschijnt het tweede plaatje, als nu beide knoppen INP1 en INP2 ingedrukt worden verschijnt het derde plaatje.

Als je een ander converter neemt kan je meer plaatjes laten zien, maar daar kan je zelf wat mee experimenteren natuurlijk.

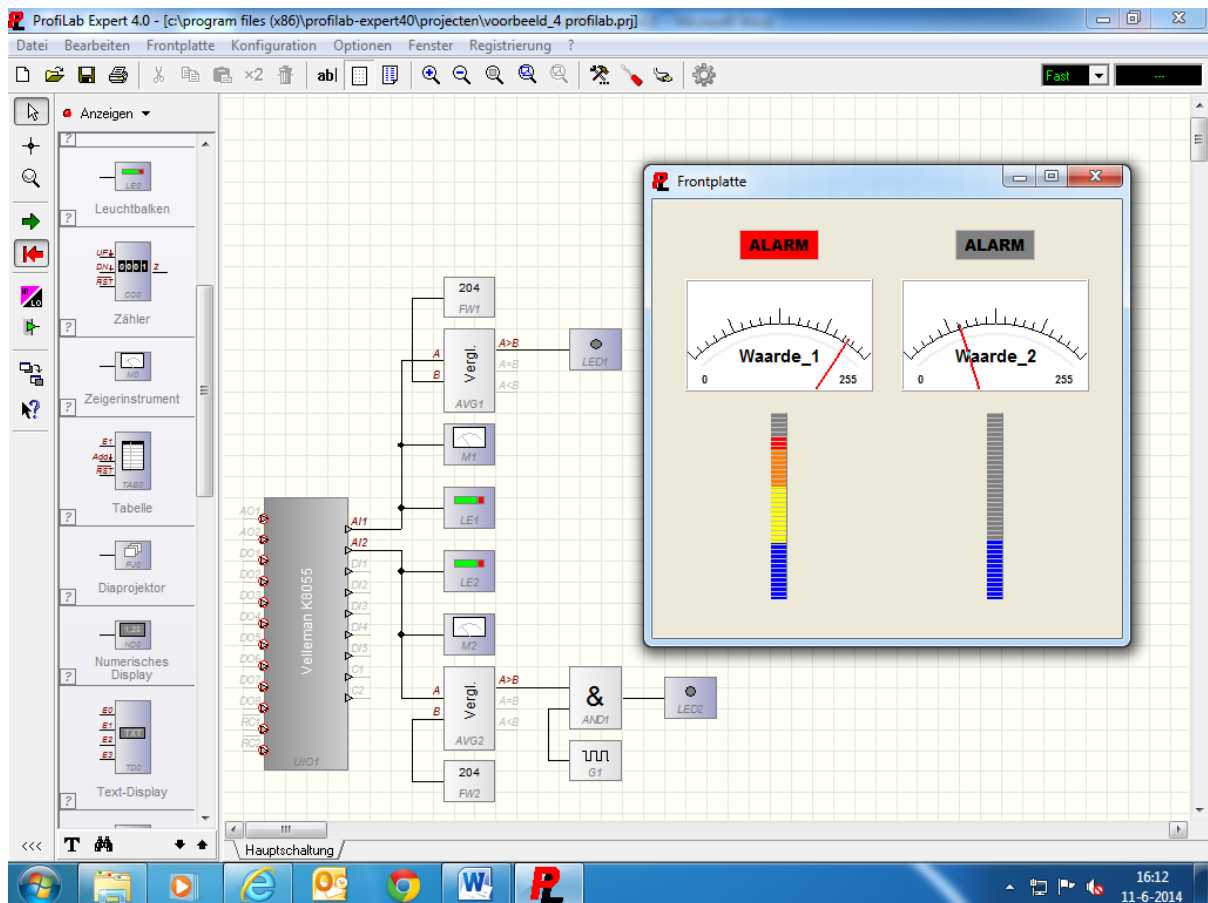
4.4: Analoge ingang

We gaan weer een nieuwe schakeling opzetten rond de K8055, we gaan iets met de analoge ingangen doen van de K8055.

Zoals al eerder gezegd is analoog niet weg te denken in de elektronica, vooral in de besturingstechniek en meet en regeltechniek. We gaan nu wat meer componenten gebruiken zoals op het voorbeeld te zien is. Zie Blz. 25.

We slepen twee vaste waarde modules, twee vergelijkers, twee analoge meters, twee ledbars, twee leds, één and poort, en één pulsgever naar het werkblad. De vaste waarde modules staan onder **analoges** en dan **quellen**. De vergelijkers staan onder **analoges** en dan **vergleicher**. De analoge meters, ledbars, en leds staan onder **frontplattenelementen** en dan **anzeigen**. De and poort staat onder **logikbauteile** en dan **gatter**. En de pulsgever staat onder **logikbauteile** en dan **zeitgeber**.

gaan alles weer verbinden zoals op het voorbeeld te zien is. Het vaste waarde moduul zetten we op de B_PIN van de vergelijker, de led zetten we op de A>B_PIN van de vergelijker. De analoge ingang AI1 komt op de A_PIN van de vergelijker, één led, en één ledbar. Nu hebben we het bovenste gedeelte aangesloten. Met het onderste gedeelte doen we het zelfde, en dit gedeelte zetten we op analoge ingang AI2. Het onderste gedeelte bevat twee extra componenten, een and poort en een pulsgever. De A>B_PIN zit op de bovenste pin van de poort, en op de onderste pin zit de pulsgever. Op de uitgang van de poort zit de led



We gaan de componenten weer instellen. We beginnen met de K8055. Als we onder eigenschappen kijken zien we links onderin het venster een knopje staan, als we hier op drukken wordt er een nieuw venster geopend. Daar selecteren we **analogeingang 1** en dan **configurieren**, we zetten nu de minimale waarde op 0 en de maximale waarde op 255. Daarna sluiten we de venster met OK. Op de print van de K8055 moeten we de twee jumpers SK2 en SK3 maken, dit doen we om straks gebruik te maken van de twee potmeters die op de print staan.

We gaan nu de vaste waarde module instellen onder eigenschappen, bij **wert** zetten we 204 neer. Het zelfde doen we met de andere module.

We gaan nu de analoge meter instellen. Daarvoor gaan we naar de frontplaat onder eigenschappen kan je een waarde ingeven bij **bereich**. We zetten de minimale waarde op 0 en de maximale waarde op 255. Bij **einheid** zetten we waarde_1 neer. Met de rest van de instellingen doen we niets. Het zelfde doen we bij de andere analoge meter.

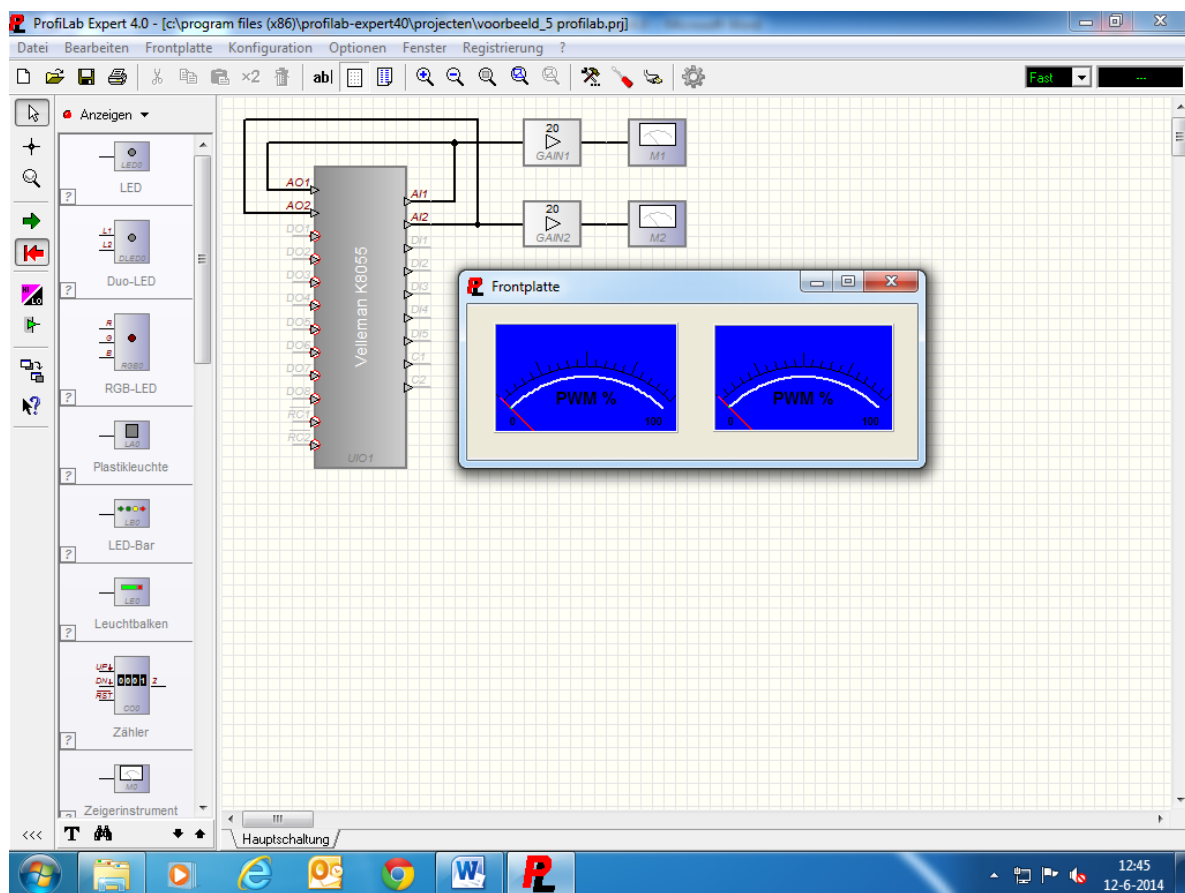
We gaan nu de ledbar instellen. Onder eigenschappen zetten we het **bereich** van 0 tot 255, **farbe 1** maken we blauw, **farbe 2** maken we geel, **farbe 3** maken we groen, en **farbe 4** maken we rood. Nu zetten we **grenze 1 [%]** op 30, **grenze 2[%]** op 60, en **grenze 3 [%]** op 80. De rest van de instellingen laten we zo staan. Het zelfde doen we bij de andere ledbar.

Nu stellen we de leds in. Voor de leds kiezen we een rechthoek, met de tekst **ALARM** erin, voor de kleur nemen we rood. We maken de leds weer zo groter, zodat de tekst goed past.

We gaan nu de frontplaat netjes indelen, als dat gebeurt is gaan we naar het werkblad. We moeten nog één component instellen, en dat is de puls geveer. Onder eigenschappen geven we de waarde twee in, de puls geveer geeft nu twee pulsen per seconden.

We gaan nu de simulatie starten. Als we nu aan potmeter ATT1 draaien zal de ledbar oplopen en van kleur veranderen, en de analoge meter geeft de stand aan tussen de 0 en 255. Als de waarde nu boven de 204 komt zal de led gaan branden, en het bovenste deel van de led bar wordt rood. Als we nu aan potmeter ATT2 draaien gebeurt precies het zelfde, alleen als de waarde boven de 204 komt gaat de led niet continue braden maar knipperen met de frequentie die de puls geveer afgeeft.

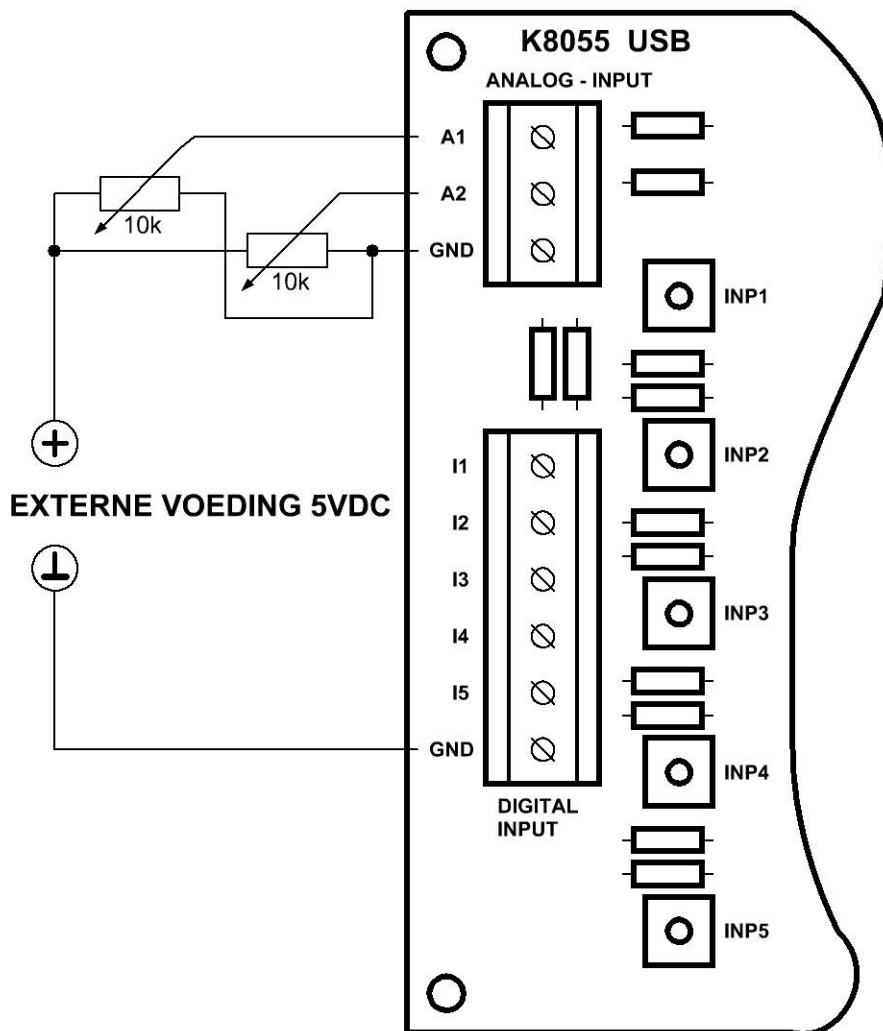
4.5: Analoge uitgang



We gaan nu wat doen met de analoge uitgangen van de K8055. We slepen twee versterkers, en twee analoge meters naar het werkblad. De versterkers staan onder **analoges** en dan **verstärker**. We gaan de analoge ingangen en de analoge uitgangen koppelen met elkaar. Dus AO1 en AI1, en AO2 en AI2 komen aan elkaar. Dit doen we om straks met de printpotmeters op de kaart de analoge uitgangen te kunnen bedienen. Op de kaart zijn de analoge uitgangen en de PWM uitgangen gekoppeld aan elkaar, dus als de analoge uitgangen aangestuurd worden, worden ook de PWM uitgangen aangestuurd. AO1 en AI1 gaan naar de ingang van versterker_1, en AO2 en AI2 gaan naar de ingang van versterker_2. De uitgang van versterker_1 gaat naar de ingang van meter_1, en de uitgang van versterker_2 gaat naar de ingang van meter_2.

We gaan nu de componenten instellen. De K8055 zetten we op een waarde tussen de 0 en 5, dat doen we zowel bij de analoge in als uitgangen.

Dit kan weer onder eigenschappen en daarna klikken we op het knopje links onderin het venster. Daar selecteren we **analogeingang 1** en **analogeingang 2**, en zetten de waarde onder **konfigureren** minimaal op 0 en maximaal op 5. Dit herhalen we ook voor de analoge uitgang. De versterkers stellen we in onder eigenschappen, daar zetten we onder **versterking** het getal 20 neer. Hetingangssignaal zal nu 20 X versterkt worden, dit doen omdat we om de meters straks een waarde van 0 tot 100 aan moet geven. We gaan de meters instellen. In dit geval krijgen de meters een blauwe achtergrond en een waarde tussen de 0 en 100, onder **eenheid** zetten we nu PWM %. Onder **trägheid** laten we de waarde op 10 staan. Dit kan allemaal weer onder eigenschappen.



Als we de simulatie opstarten zien we dat als er aan de printpotmeters op de K8055 gedraaid wordt dat de leds bij PWM1 en PWM2 feller gaan branden, ook zien we de meters uitslaan van 0 tot 100 de meters geven het PWM percentage aan. PWM signalen dienen met name voor het bedienen van dimmers, en het aansturen van elektromotoren. Maar daar over later meer.

We kunnen dit ook doen doormiddel van externe potmeters, de potmeters kunnen we aansluiten op de printkroonstenen van de K8055. Als we dit doen moeten we de jumpers SK2 en SK3 verwijderen, op deze manier kan je met externe potmeters werken, ook moeten de potmeters op de print ATT1 en ATT2 helemaal naar rechts gedraaid worden. Als dat niet gedaan wordt kan het uitgangssignaal niet voor de volle honderdprocent uitgestuurd worden. Hier onder nog een schema hoe je zoiets aan kan sluiten.

De potmeters komen met de lopers aan A1 en A2. Eén uiteinde van de potmeters komt aan de GND te zitten, en de andere uiteinden van de potmeters komt aan de voeding te zitten. Je kan als voeding een netadapter gebruiken, dat mag ook een geschakeld exemplaar zijn.

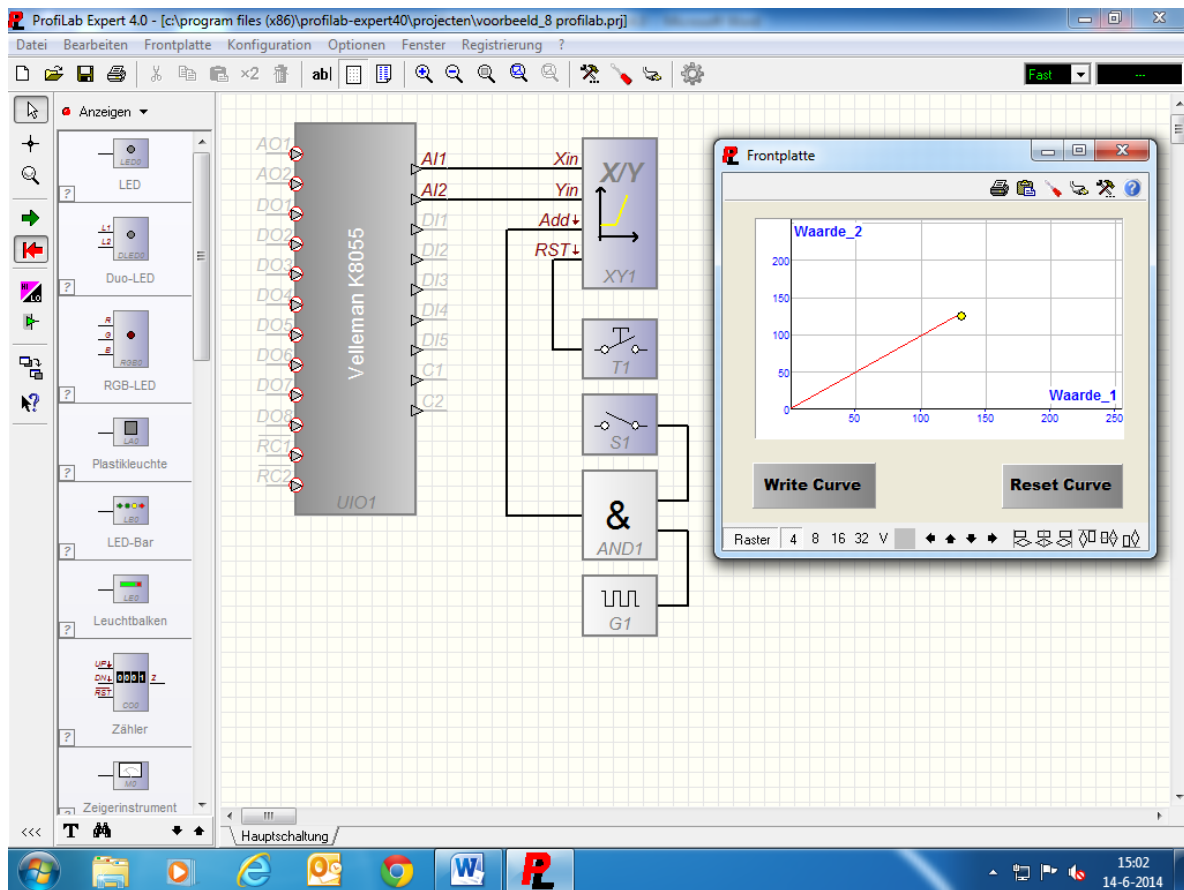
De voeding externe voeding kan eventueel ook uit de pc gehaald worden, hiervoor kan je bijv. de spanning uit de usb poort gebruiken. Hier kan je zelf een kabeltje voor maken, of er één kopen.

Zoals op het schema te zien is op Blz. 27, worden er twee verschillende ground (GND) aansluitingen gebruikt. Dat kan en mag omdat alle Ground aansluitingen op de print aan elkaar hangen.

4.6: X/Y schrijver

Ik ga vanaf nu niet meer vertellen waar alles staat. Ook de instellingen ga ik niet meer uitleggen, alle instellingen gaan via eigenschappen dat kan op het werkblad maar kan ook op de frontplaat zijn. Ik ga alleen opnoemen wat we nodig hebben, en wat de waardes van de instellingen moeten zijn die eventueel in de modules moeten komen.

We slepen een X/Y schrijver, een drukknop, een schakelaar, een and poort, en een pulsgever naar het werkblad. De analoge ingangen AI1 en AI2 komen aan Xin en Yin van de schrijver te zitten, de drukknop komt op RST en de uitgang van de and poort komt op de Add ingang te zitten. De schakelaar en de pulsgever komen aan de ingang van de and poort te zitten. We gaan alle componenten weer instellen. We beginnen met de K8055, we zetten de waarde van de analoge ingangen AI1 en AI2 van 0 tot 255, en we zetten de jumpers SK2 en SK3 weer op de print. Nu stellen we de X/Y schrijver in, onder bereik zetten we de waarde ook van 0 tot 255. Dat doen we zowel bij de X als de Y as, bij eenheid geven we de tekst **Waarde_1** en **Waarde_2** in. Voor de kleuren kiezen we wit voor de achtergrond, grijs voor het raster, zwart voor de nullijn, en blauw voor de tekst. Voor de drukschakelaar en de schakelaar kiezen we weer voor een type met tekst. De schakelaar geven we de tekst **Write curve** mee, en de drukknop geven we de tekst **Reset curve** mee. Als laatste geven we de pulsgever een waarde van 50 mee, op deze manier wordt de ingelezen waarde 50 maal per seconden op het scherm gezet. Zie voorbeeld op Blz. 29.

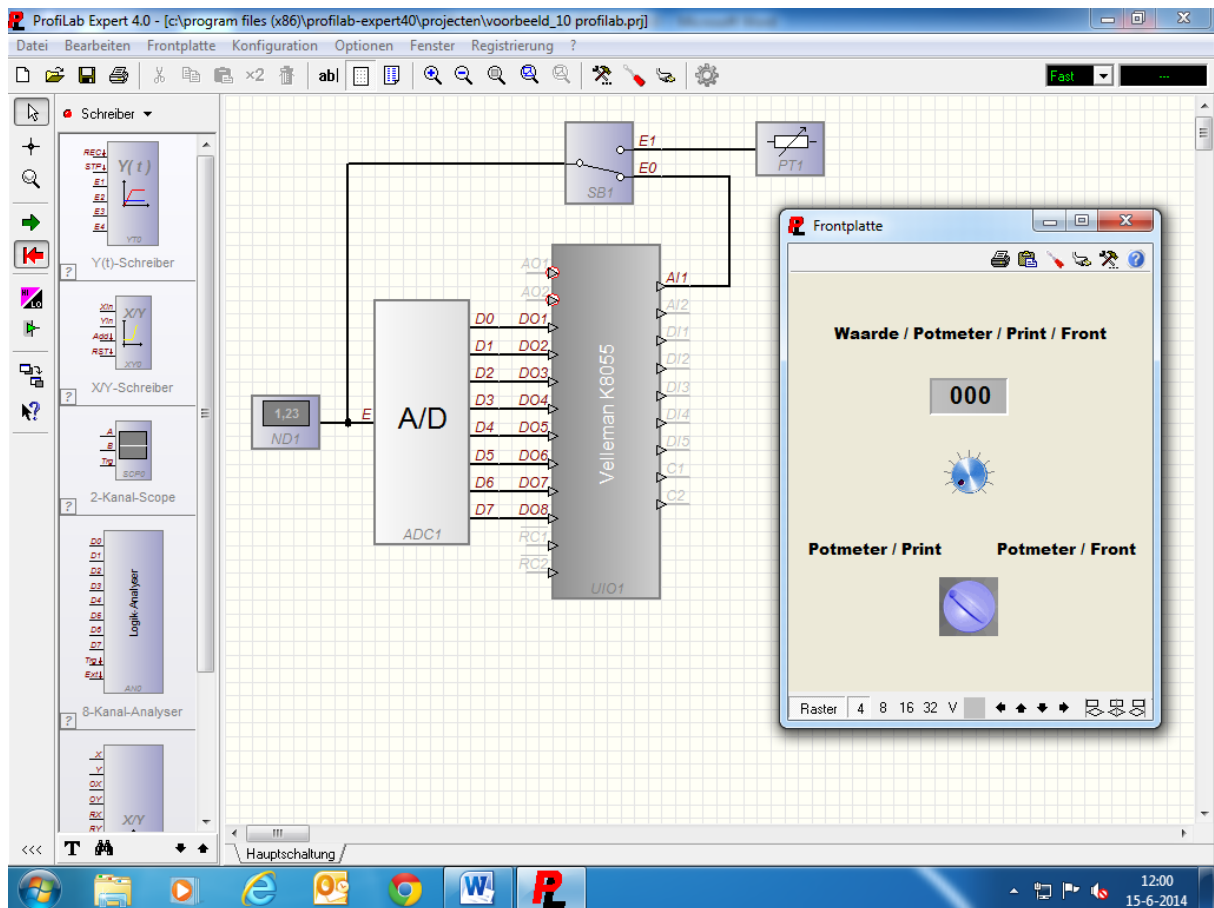


Als we nu de simulatie starten en de Write curve knop indrukken, zal je zien als we aan de printpotmeters draaien dat er een lijn op de schrijver zichtbaar wordt. Als we op de knop *Reset curve* drukken wordt de lijn gewist. We hebben nu de waarde 0 tot 255 ingegeven bij de K8055 en de schrijver, als je die waarde groter maakt bijv. 0 tot 1024 kan je weer andere grootheden meten bijv. luchtdruk. Je kan aan de analoge ingangen bijv. ook een temperatuur sensor hangen, **zie schema analoge ingangen van de K8055** de potmeters worden dan vervangen door een analoge temperatuursensor, op deze manier kan je dan een temp schrijver maken.

4.7: Schakelen tussen potmeter print en front

We blijven nog even bij de analoge functie van de K8055. We gaan nu schakelen tussen twee potmeters, de potmeter die op de print zit en de potmeter die op de frontplaat staat. Zie voorbeeld op Blz. 28.

We slepen een display, een A/D converter, een analoge schakelaar, en een potmeter naar het werkblad. Voor de A/D converter nemen we een 8 bit type, D0 van de converter komt aan DO1 van de K8055, D1 van de converter komt aan DO2 van de K8055, enz. enz. Aan de ingang E van de converter komt het display en de uitgang van de analoge schakelaar te zitten. Aan de ingang van de analoge schakelaar komt de potmeter die op de frontplaat staat op E1, en op E1 komt de analoge ingang AI1 van de K8055. Nu we alles weer verbonden hebben gaan we de componenten weer instellen. We beginnen met de K8055.

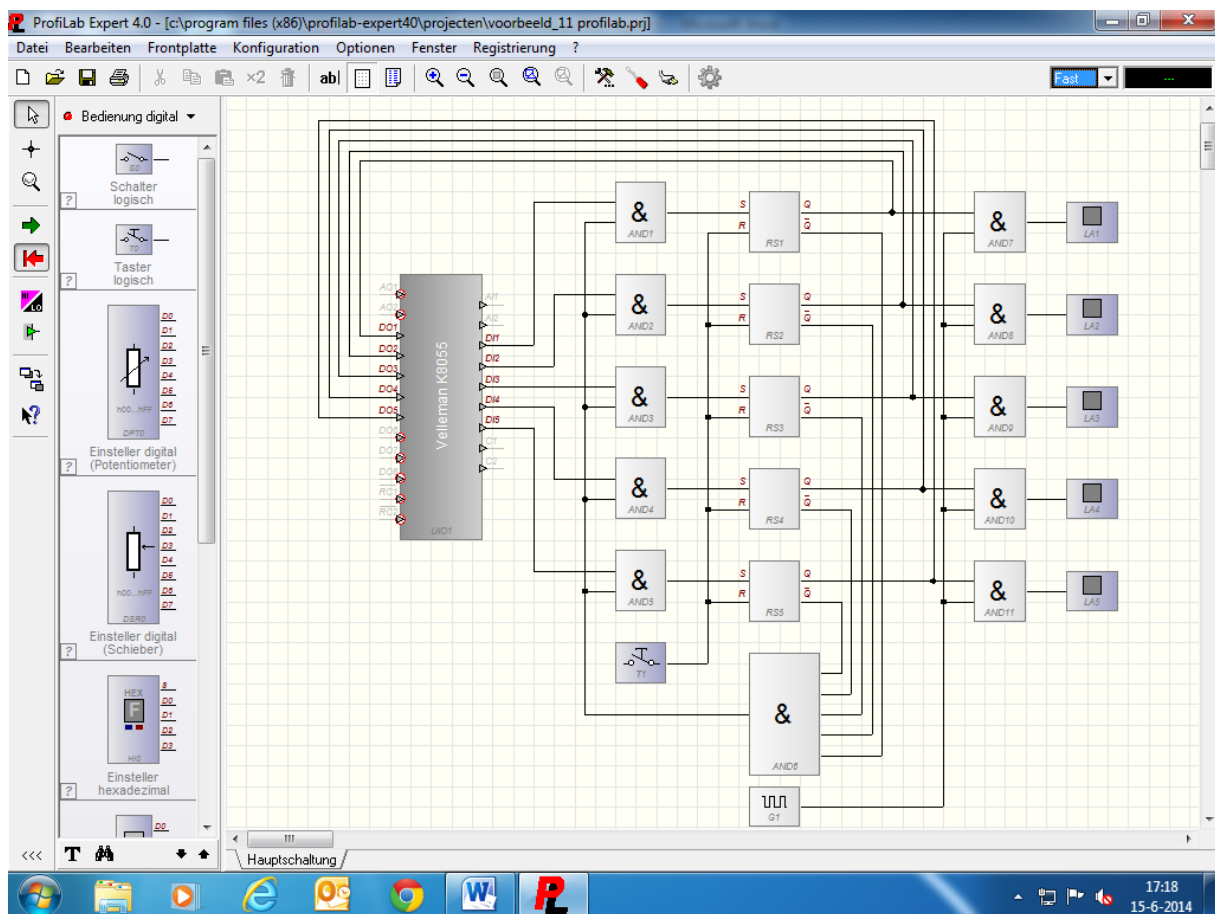


De analoge ingang AI1 zetten we op een waarde van 0 tot 255. De A/d converter staat op 8 bit, en de waarde stellen we in op 0 tot 255. Voor de analoge schakelaar kiezen we **draaischakelaar 2**, en de knop maken we blauw.

Het bereik van de potmeter stellen we in op 0 tot 255, en ook daar maken we de knop ook blauw. Het aantal stappen van de potmeter zetten we ook op 255, zo kunnen we het hele bereik van de potmeter benutten. Het display zetten we op drie digit, en de komma zetten we vast. Nu plaatsen we nog wat tekst op de frontplaat. Boven het display komt *Waarde Potmeter / Print / Front* Links boven de schakelaar komt *Potmeter / Print*, en rechts boven de schakelaar komt *Potmeter / Front*.

We gaan nu de simulatie starten. Als de schakelaar op potmeter / print staat, kan je de schakeling met de printpotmeter bedienen. Als de schakelaar op potmeter / front staat, kan de schakeling via de potmeter op het front bediend worden. Als er aan één van de potmeters gedraaid wordt zullen de uitgangen van de K8055 vollopen, dit gebeurt weer binair. Het display zal de decimale waarde aan geven. De printpotmeter kan ook hier weer uitgeschakeld worden door de jumper bij ATT1 Te verwijderen, en de potmeter weer helemaal naar rechts te draaien. Nu kan er weer een externe potmeter op de K8055 aangesloten worden. **zie schema analoge ingangen van de K8055**

4.8: Quiz tafel



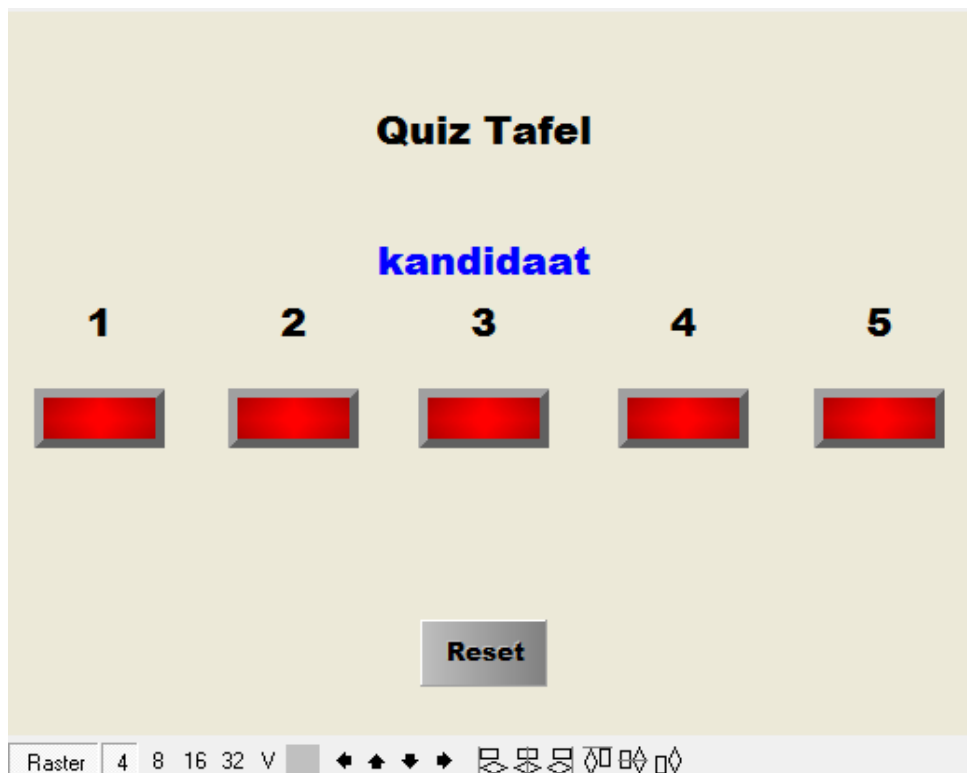
We gaan een quiz tafel maken. We slepen elf and poorten, vijf flipflops, vijf signaal lampen, en één pulsgever naar het werkblad. We gaan nu alles met elkaar verbinden. De digitale ingangen van de K8055 DI1 t/m DI5 komen aan de eerste vijf and poorten AND1 t/m AND5, aan de bovenste ingang. De onderste ingangen komen aan de uitgang van de vijf poorts and poort. De uitgangen van de eerste vijf and poorten komen aan de set ingang van de Flipflops, de reset van de flipflops zitten aan de reset drukknop. De geïnverteerde Q uitgangen van de flipflops komen aan de ingang van de vijf poorts and poort. De niet geïnverteerde Q uitgangen van de flipflops komen aan de bovenste ingangen van de and poorten AND7 t/m AND11.

Het knooppunt van die aansluitingen komen aan de digitale uitgangen DO1 t/m DO5 van de K8055. De onderste aansluitingen van de and poorten AND7 t/m AND11 komen aan de uitgang van de pulsgever. De uitgangen van and poorten AND7 t/m AND11 komen aan de lampen LA1 t/m LA5. Als het goed is dan is alles nu met elkaar verbonden. We gaan nu de componenten instellen. De pulsgever zetten we op twee pulsen per seconden. We gaan nu naar de frontplaat om de lampen en de drukknop in te stellen. We kiezen voor een rechthoekig model, en de lenzen maken we rood. De drukknop geven we de tekst Reset mee. Nu zetten we nog wat tekst op de frontplaat. Boven aan de frontplaat komt Quiz Tafel in het zwart, daar onder komt Kandidaat in het blauw, en boven de lampen komt 1 t/m 5 in het zwart.

Als alles goed is komt het er dan zo uit te zien, zie voorbeeld hieronder.

We gaan nu de simulatie starten.

Als er een vraag gesteld is en er wordt door de snelste kandidaat op één van de knoppen gedrukt van de K8055, dan zal de lamp op de frontplaat gaan knipperen. Dat knipperen wordt geregeld door de pulsgever. Als er één maal een lamp knippert worden de andere knoppen geblokkeerd door de schakeling. Dit komt door de flipflops in combinatie met de and poorten AND1 t/m AND5, en de vijf poorts and poort AND6.



Als er op de resetknop wordt gedrukt staat de schakeling weer klaar voor de volgende ronde.

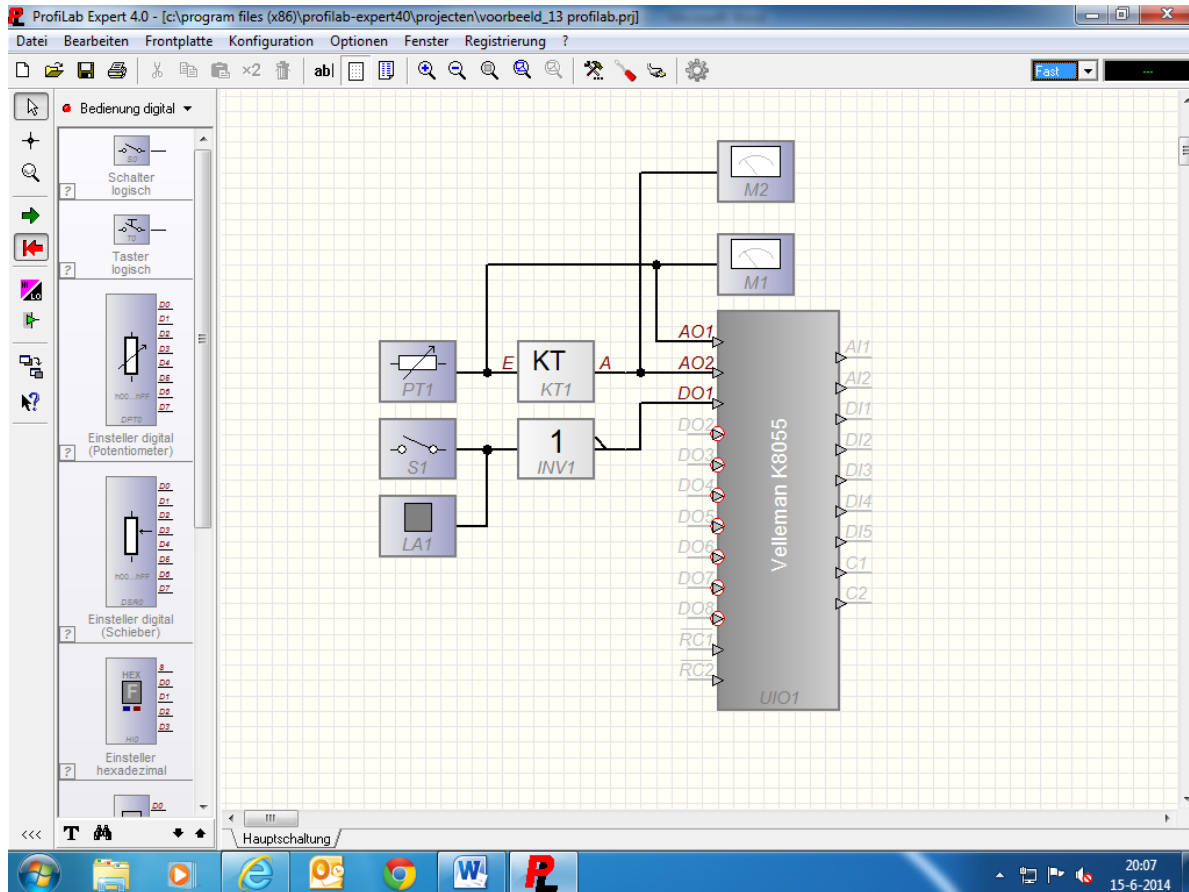
Er kunnen maar vijf kandidaten mee doen, omdat we niet meer ingangen hebben op de K8055.

Maar je krijgt wel een idee hoe je zoiets kan maken. Ook is er de mogelijkheid om er externe drukknoppen op te maken, dat kan weer op de zelfde manier als bij het **Zie schema digitale ingangen van de K8055** De lampen kunnen ook extern uitgevoerd worden, die kan je dan weer aansluiten op de uitgangen van de K8055. Dat kan rechtstreeks, maar kan bijv. ook via relais.

Zie schema digitale uitgangen van de K8055

De schakeling kan ook gemaakt worden met andere interface kaarten. Als je bijv. de interface van Hygrosens neemt, **Hygrosens usb 32 in en 32 out** Dan heb je 32 in en 32 uitgangen tot je beschikking. Daar kan je dus heel wat mee doen. De interface kan direct op de usb poort aangesloten worden.

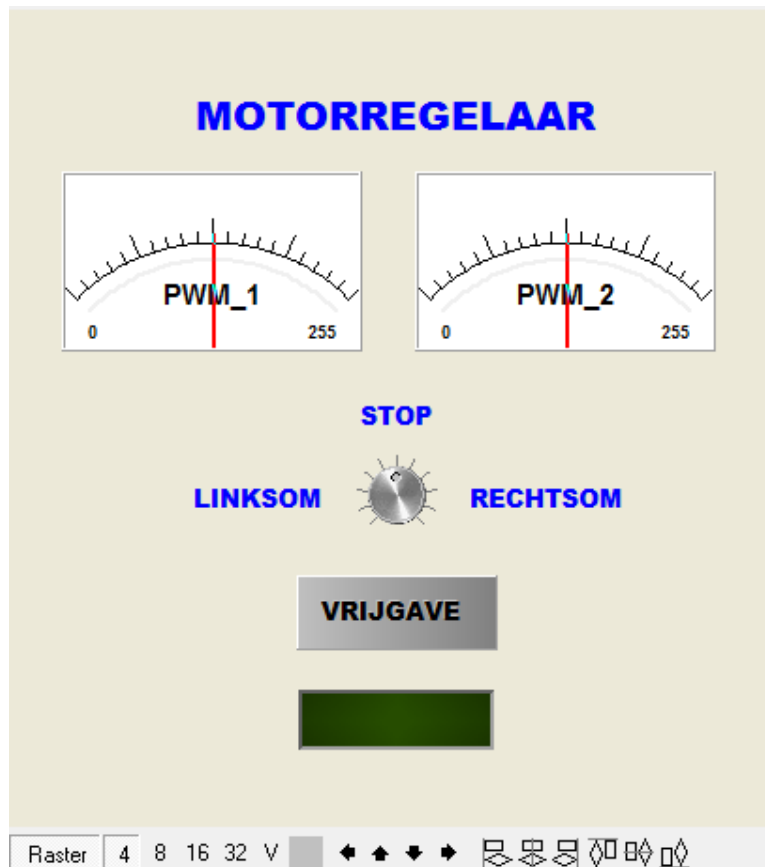
4.9: Motorregelaar



We gaan een besturing voor een motorregelaar maken. We slepen twee meters, één potmeter, één schakelaar, één lamp, één correctie module, en één inverter naar het werkblad. We gaan nu alles weer verbinden met elkaar. De potmeter komt aan de ingang van de correctie module, aan de meter M1, en aan analoge uitgang AO1 van de K8055. De uitgang van de correctie module komt aan meter M2, en aan analoge uitgang AO2 van de K8055. De schakelaar komt aan lamp LA1 en aan de ingang van de inverter. De uitgang van de inverter komt aan digitale uitgang DO1. We hebben nu alles weer aangesloten, en nu gaan we de componenten weer instellen. De waarde van de potmeter zetten we van 0 tot 255, en het bereik van de potmeter zetten we ook op 255. De voor in gave van de potmeter zetten we op 127. Bij de correctie module zetten we op de bovenste regel onder ingang 0 en onder uitgang 255. Op de regel daar onder zetten we onder ingang 255 en onder uitgang 0. Op deze manier wordt het potmeter signaal geïnverteerd. De meters zetten we beide op een waarde van 0 tot 255. De tekst in meter M1 wordt PWM_1, en in meter M2 wordt dat PWM_2. De schakelaar voorzien we van de tekst VRIJGAVE. De lamp maken we groen.

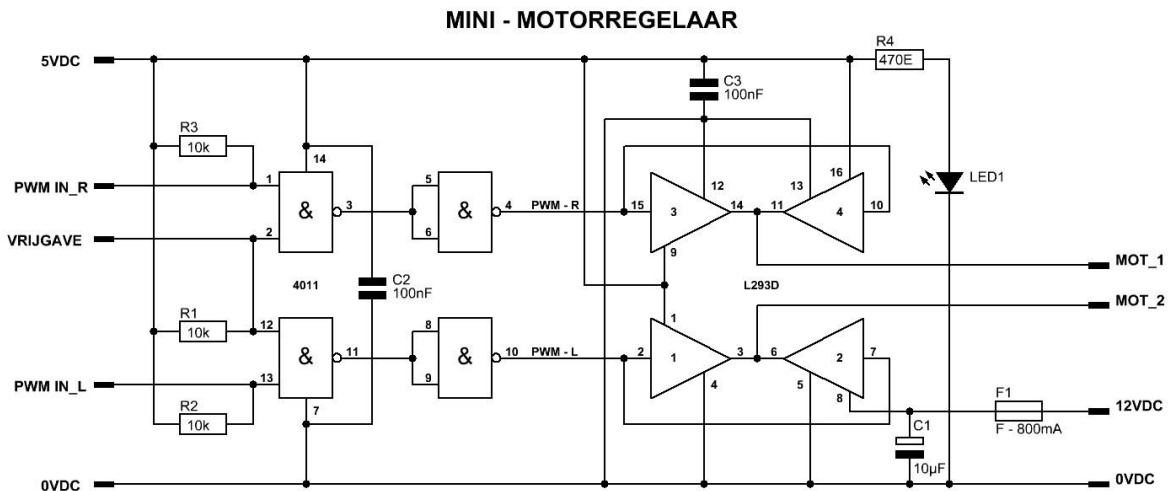
We gaan nu nog wat tekst op de frontplaat zetten. Boven de meters zetten we **MOTORREGELAAR**, boven de potmeter zetten we **STOP**, links van de potmeter zetten we **LINKSOM**, en rechts van de potmeter zetten we **RECHTSOM**. Alle tekst wordt hier in het blauw neer gezet.

En als we alles goed gedaan hebben komt de frontplaat er zo uit te zien.



We gaan de simulatie weer starten. Als de simulatie gestart is zal je zien dat beide meters op de helft staan, ook de beide leds bij de pwm uitgangen branden. Die branden echter op halve sterkte omdat de potmeter in de middenstand staat. De waarde is nu dus 127. Als we de potmeter nu linksom draaien zal je zien dat de meter met PWM_1 naar de nul stand gaat, en dat meter PWM_2 naar de stand 255 gaat. Als we rechtsom draaien gebeurt het tegenover gestelde. Dat signaal gaan we straks gebruiken voor het aansturen van de motorregelaar Ook zie je nu dat de led bij de eerste uitgang brand, als er nu op vrijgave gedrukt wordt gaat de led bij de uitgang uit en de lamp op de frontplaat gaat aan. Dit wordt straks de vrijgave van de motorregelaar.

We gaan straks de L293D als motorregelaar gebruiken, hier kan je een kleine H_brug mee maken om motoren in toerental en richting te regelen. Hier het schema van de regeling rond de L293D.



Zoals te zien is op het schema staan er twee eindtrappen parallel, (rechter gedeelte schema) dit is gedaan zodat de regelaar meer stroom kan leveren. Aan de linkerkant zien we een 4011 staan dit is een nand poort. De linkse twee poorten staan als and not, en de rechtse twee poorten staan als inverter geschakeld.

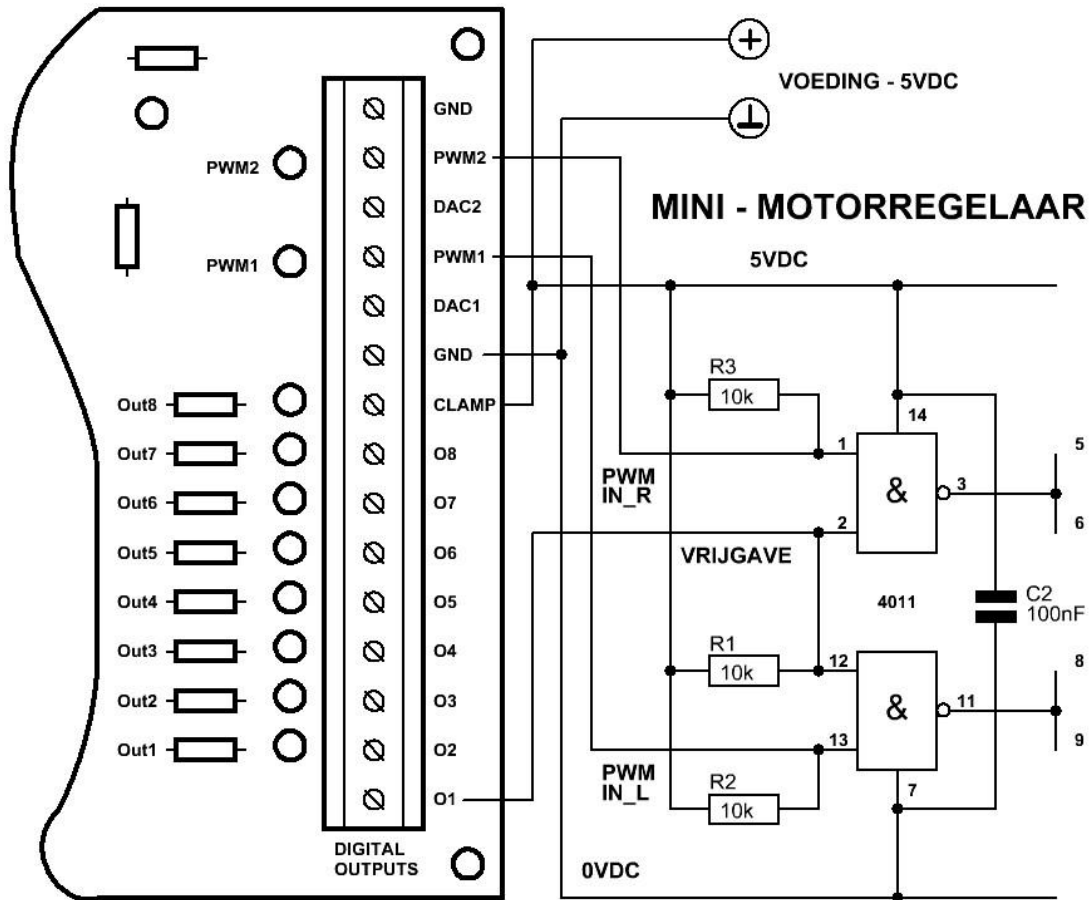
We sluiten nu de vrijgave pin aan op de digitale uitgang DO1. En de twee pwm uitgangen PWM1 en PWM2 sluiten we aan op de pwm pinnen van de regelaar. De 0VDC aansluiting zetten we op een GND aansluiting van de K8055. De 5VDC aansluiting van de motorregelaar komt aan de clamp aansluiting van de K8055, en daar sluiten we ook een externe 5VDC voeding op aan. De 0VDC van de voeding komt weer aan de GND van de K8055. Als voeding kan een netadapter gebruikt worden.

Aan de uitgangskant van de motorregelaar sluiten we een motor aan op de pinnen MOT_1 en MOT_2, voor de motor kiezen we bijv. een type van 12VDC. Op de voeding pinnen sluiten we ook een externe voeding aan van 12VDC. Ook dit kan weer een netadapter zijn.

We hebben gezien dat als we de simulatie starten de led bij de uitgang DO1 brand, dat betekent dat de uitgang laag is. Dus de vrijgave pin is nu ook laag, de pwm signalen komen nu niet door de nand poorten heen. De motor zal dus stil blijven staan. Als we nu de vrijgave knop indrukken wordt de vrijgave pin van de motorregelaar hoog. Als we nu aan de potmeter draaien zal de motor gaan draaien, hoe verder we de potmeter verdraaien hoe sneller de motor gaat draaien. Als we de potmeter naar links draaien moet de motor ook linksom gaan draaien, als dat niet het geval is moeten de twee aansluitingen PWM_1 en PWM_2 omgedraaid worden.

Je kan hier een print voor maken, of je bouwt de schakeling rond de motorregelaar op een bread board. Het kan ook op gaatjes board natuurlijk.

Op Blz. 36 zie je hoe je het op de K8055 aan kan sluiten.

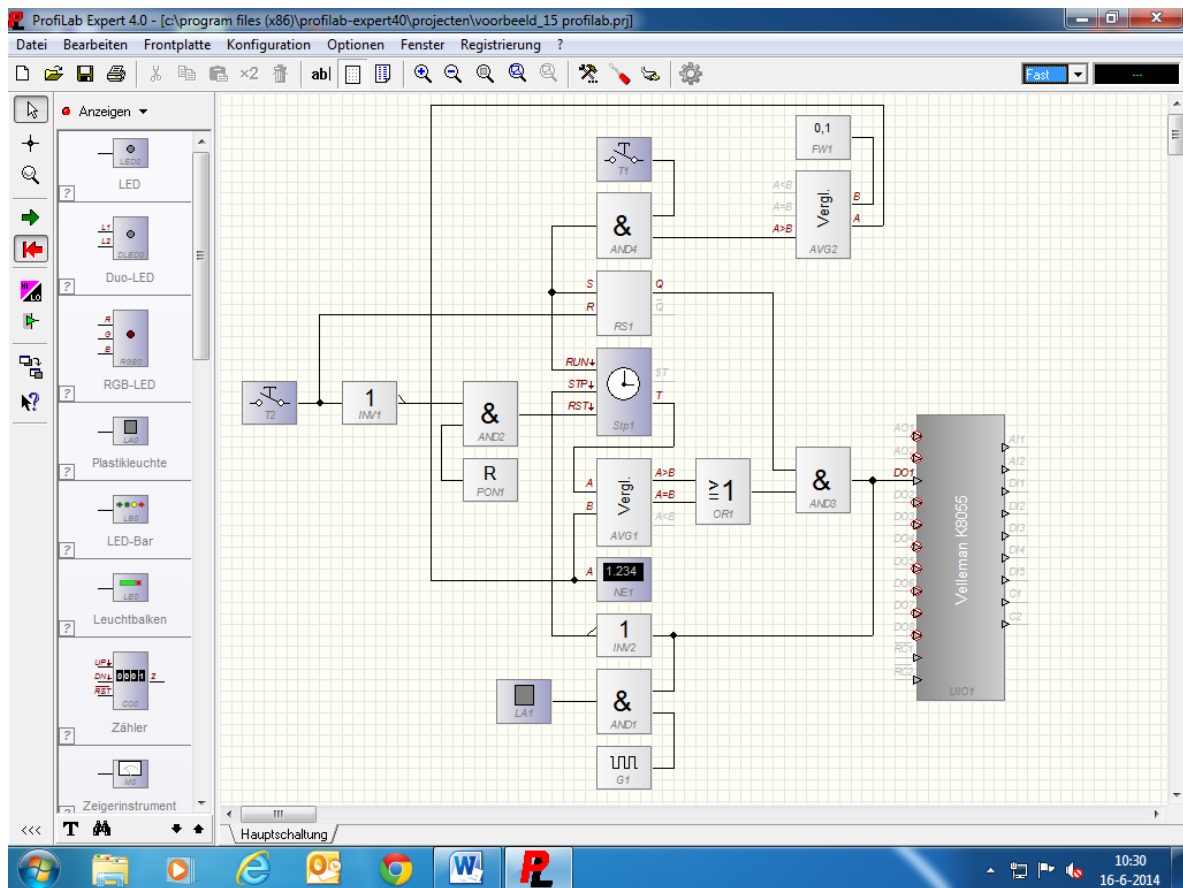


De voedingspanning voor de motor komt van een externe voeding. Dat kan bijv. een 12VDC spanning zijn.

4.10: Timer opkom vertraag

We gaan een timer maken die opkomend vertraagt gaat werken. We slepen twee drukknoppen, twee inverters, vier and poorten, één or poort, twee vergelijkers, één vaste waarde module, één flipflop, één power on reset module, één klok module, één numeriek ingave module, één lamp, één pulsgever, en één K8055 module naar het werkblad. We verbinden alles weer met elkaar, zoals op het werkblad te zien is. Met drukknop_1 kan de timer gestart worden, maar dat kan niet eerder voordat er een waarde ingegeven is op de ingave module. Dat wordt gecontroleerd door vergelijker AVG2, en and poort AND4. Als de waarde groter is dan 0.1, (die in de vaste waarde module FW1 staat) dan kan de timer gestart worden. Als de voorwaarde klopt en er wordt op start gedrukt dan wordt flipflop RS1 ge set, en wordt er één ingang van de and poort AND3 hoog. Tevens wordt ook de klok STP1 ge set en de waarde wordt vergeleken door vergelijker AVG1 met de waarde die in de ingave module NE1 staat. Aan uitgangen van de vergelijker AVG1 zit een or poort. De or poort kijkt naar $A > B$ of $A = B$, als één van deze twee waar is dan geeft de poort dat door aan de and poort AND3.

Op Blz. 37 zie je het werkblad staan.

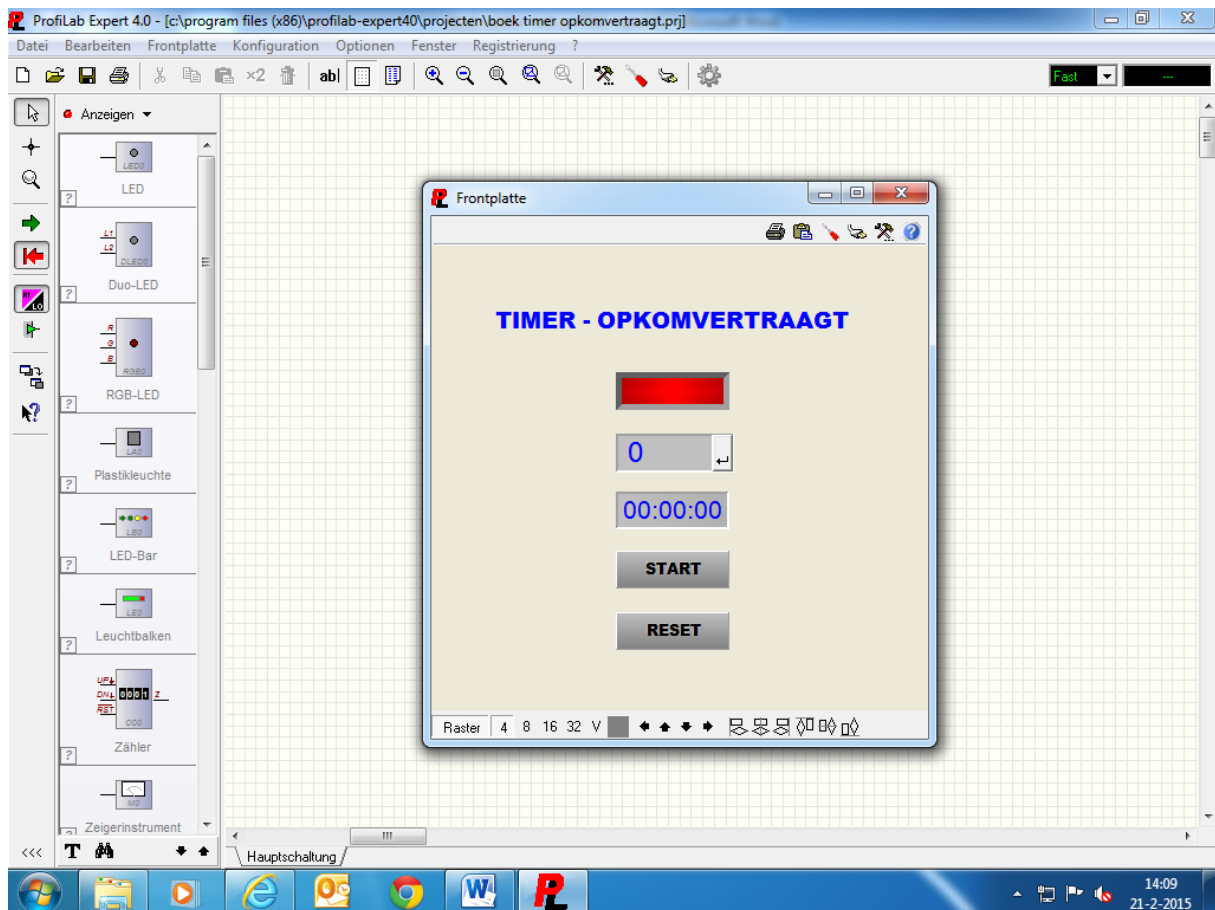


Daar was al een ingang van hoog, (door de start knop) dus de uitgang van AND3 zal nu zijn hoge signaal doorgeven aan de K8055. Ook stopt de klok STP1 met lopen doordat STP ingang op de klok module laag gemaakt wordt, dat wordt geregeld door de inverter INV2. Ook zal de lamp LA1 nu gaan knipperen dat wordt geregeld door and poort AND1 en de puls gever G1. De puls gever staat op 2Hz, en met die frequentie zal lamp LA1 ook knipperen.

Om de timer nu te resetten moet drukknop T2 ingedrukt worden, door T2 in te drukken wordt zowel de flipflop RS1 als de klok STP1 gereset. De reset voor de klok STP1 loopt via inverter INV1 en and poort AND2, dat is op deze manier gedaan zodat de klok ook gereset wordt als het programma opgestart wordt. Dat gebeurt dan weer via reset module PON1 en and poort AND2.

De timer kan tot tijden van 1 sec tot 99 uur op komend vertraagt gebruikt worden. De timer kan ook ten alle tijden gereset worden.

Als het goed is komt de frontplaat er zo uit te zien. Zie voorbeeld Blz. 38.



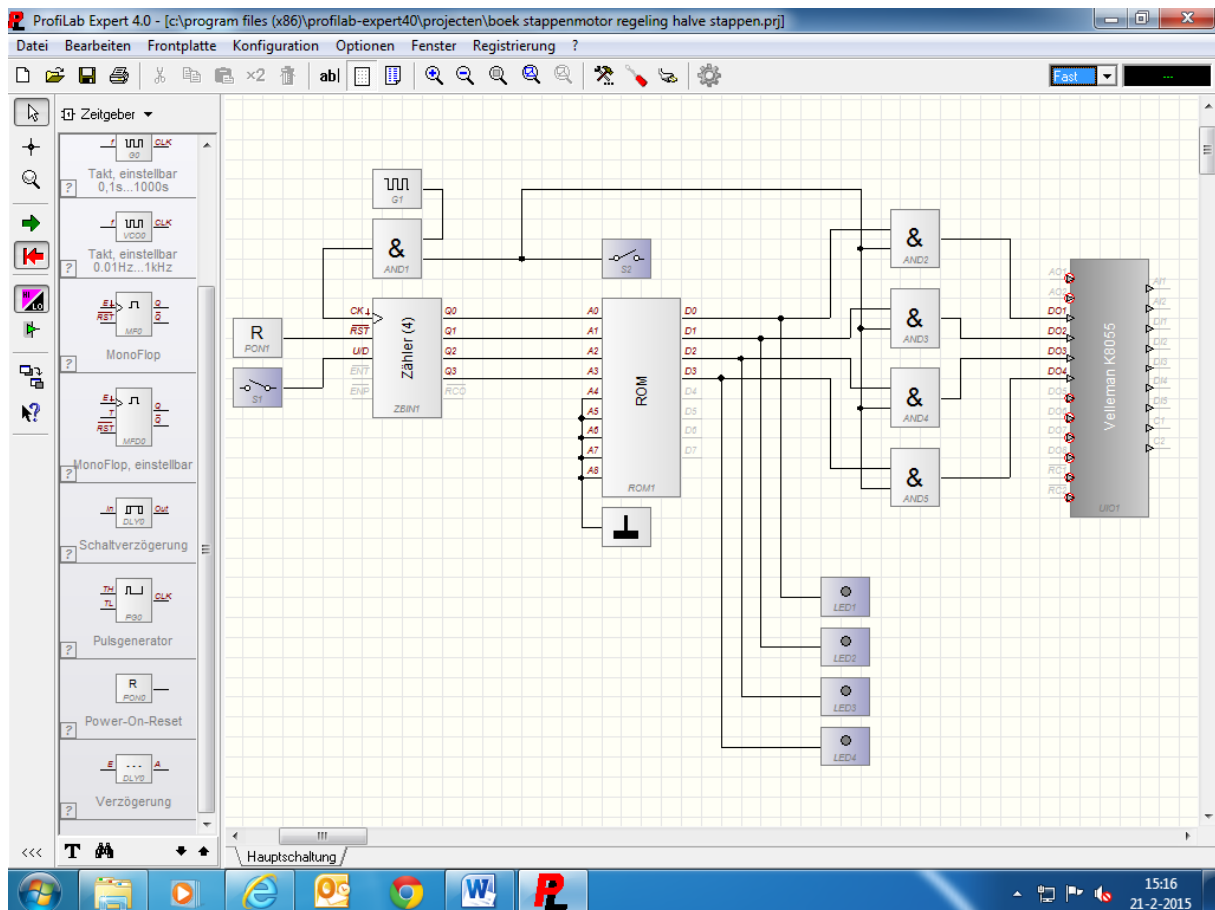
Het front van de timer komt er zo uit te zien. Van boven naar beneden de lamp, ingave module voor de tijd, de klok (geeft de verstreken tijd aan), de start, en de reset/stop knop.

Maar de frontplaat mag ook naar eigen inzicht ingedeeld worden natuurlijk.

4.11: Stappenmotorregeling

We gaan een stappenmotor regeling maken.

We slepen één drukknop, één schakelaar, één reset module, één teller module, één puls geveer, vijf maal een and poort, vier maal een led, één rom module, één massa aansluiting, en één K8055 module naar het werkblad. We verbinden alles weer met elkaar, zoals op het werkblad staat. Schakelaar S1 is een wip/steel schakelaar, met die schakelaar kunnen we de teller ZBIN1 op en af laten tellen. En met die functie kunnen we ook de stappenmotor links en rechtsom laten draaien. Met schakelaar S2 geven we de regeling vrij, en zorgen we er voor dat de teller gaat tellen. Als S2 ingedrukt wordt, wordt er één ingang van de and poorten AND1, AND2, AND3, AND4, en AND5 hoog. Poort AND1 zorg er samen met pulsgever G1 voor dat er een puls op de teller, (CK ingang komt) zodat de teller gaat lopen. De teller zorgt ervoor dat er een vier bits code op de rom module ROM1 gezet wordt, en de rom module ROM1 geeft de interne code weer door aan de and poorten AND2, AND3, AND4, en AND5. De leds LED1, LED2, LED3, en LED4 zitten op de zelfde uitgangen van de rom module om de stap volgorde te laten zien. Zie voorbeeld op Blz. 39.



De uitgangen van de and poorten zijn verbonden met D1, D2, D3, en D4 van de K8055, en daar komt dan ook de stappenmotor aan. In de rom module staat een klein stukje code voor het stap bedrijf van de stappenmotor. De code is voor half stap bedrijf, een 200 steps motor, (1.8 graden per stap) zal dus 400 stappen per omwenteling maken. Of te wel 0.9 graden per stap.

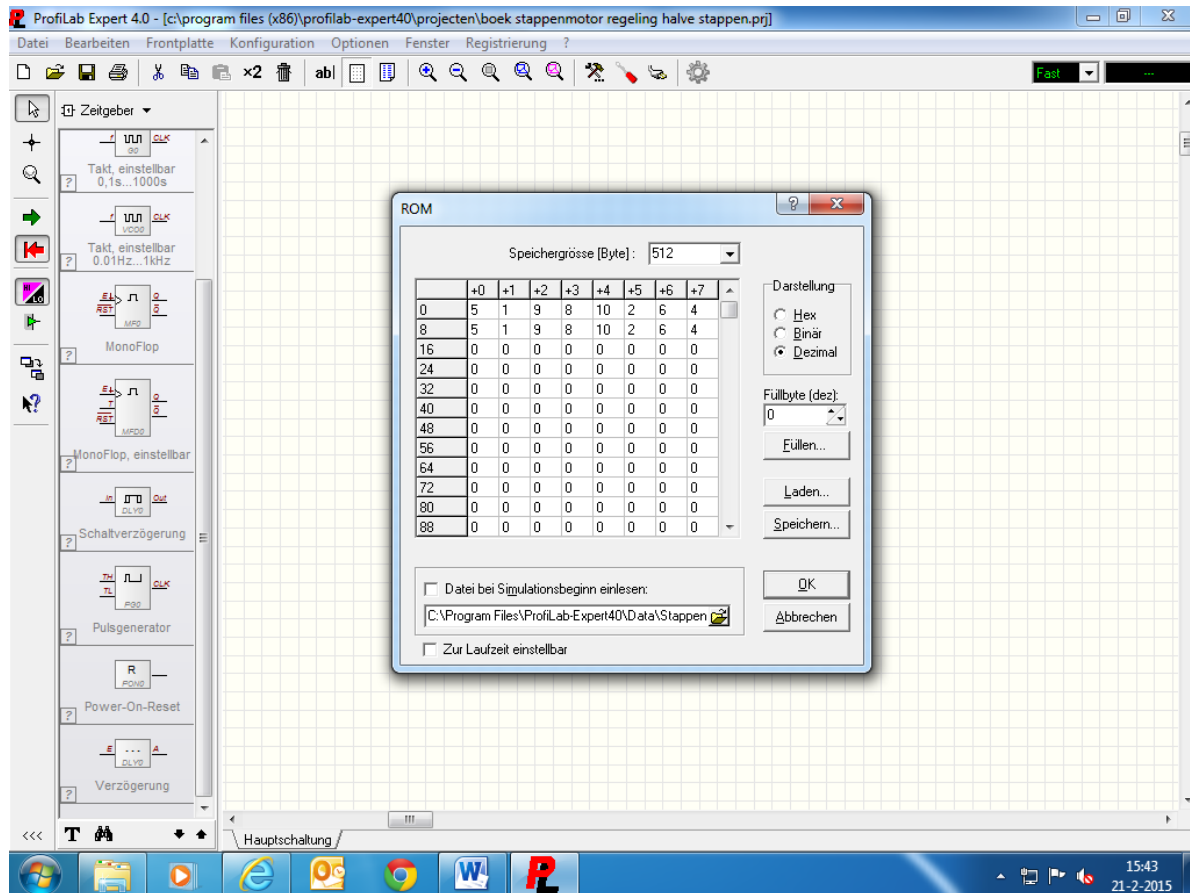
Op Blz. 40 staat het stukje code wat in de rom staat. We stellen de rom in op 512 bytes, en op de eerste twee regels komt dit te staan. De waarde is: 5, 1, 9, 8, 10, 2, 6, en 4. Dit vullen we twee maal in. De rest van de niet aangesloten pinnen A4, A5, A6, A7, en A8 van de rom module, komen aan de massa te liggen. De reset module PON1 zorgt er weer voor dat de teller op nul staat als het programma gestart wordt.

In het voorbeeld wordt er gebruik gemaakt van een puls gever module G1, de waarde van de module staat op 50Hz. Als je die waarde aanpast gaat de stappenmotor sneller of langzamer draaien, waarde groter is sneller, waarde kleiner is langzamer.

Je zou er ook voor kunnen kiezen om een de puls gever te vervangen voor een instelbare puls gever in combinatie met een potmeter module. Op deze manier is dan het toerental van de stappenmotor in te stellen via de potmeter module.

Let er wel op dat de frequentie niet te hoog gekozen wordt, omdat de stappenmotor dan stappen gaat missen. Dit is goed te horen en te zien aan de motor. Deze regeling is bedoelt voor unipolaire stappenmotoren, dat zijn motoren met vijf of zes aansluit draden.

Hieronder de code.

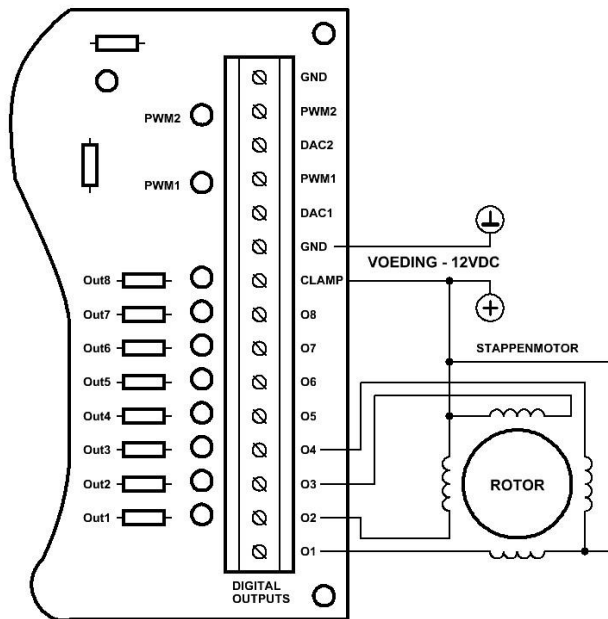


Op deze manier kan de stappenmotor op de K8055 aan gesloten worden. Ook hier zijn weer geen vrijloop diodes nodig (net zoals bij het voorbeeld met het relais) omdat die als in de ULN2803 zitten.

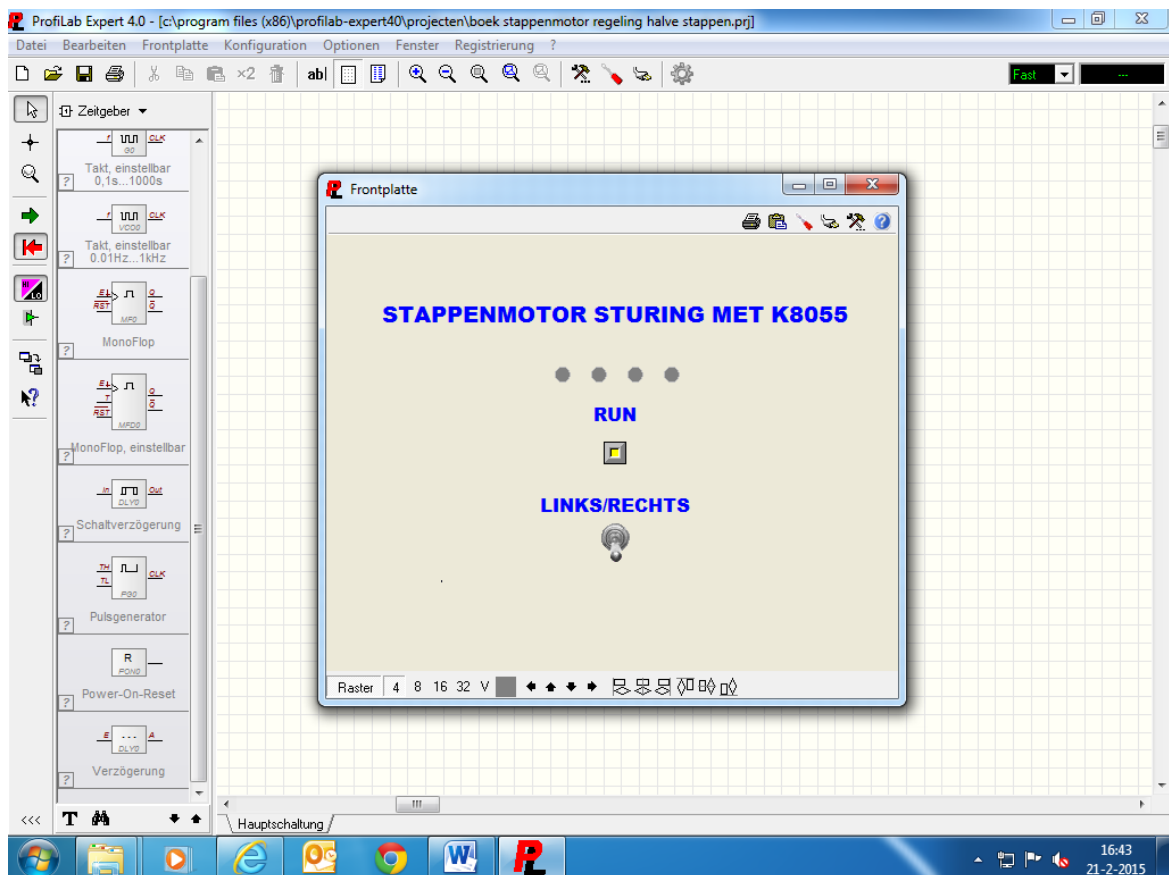
Het hier getekende voorbeeld is een unipolaire stappenmotor met zes aansluit draden. Twee draden komen aan de plus van de voeding, en de andere vier draden komen aan de uitgangen van de K8055. De kleuren en de aansluiting van de draden kan per motor verschillen, dus dat moet even proefondervindelijk uit geprobeerd worden.

De hier gebruikte spanning staat op 12VDC, maar dat kan bijv. ook 5VDC zijn. Maar dat ligt een beetje aan het type stappenmotor dat gebruikt wordt. Om er achter te komen wat de plus draden zijn, dit kan met een multimeter op standje Ohm uitgemeten worden.

Zie schema op Blz. 41.

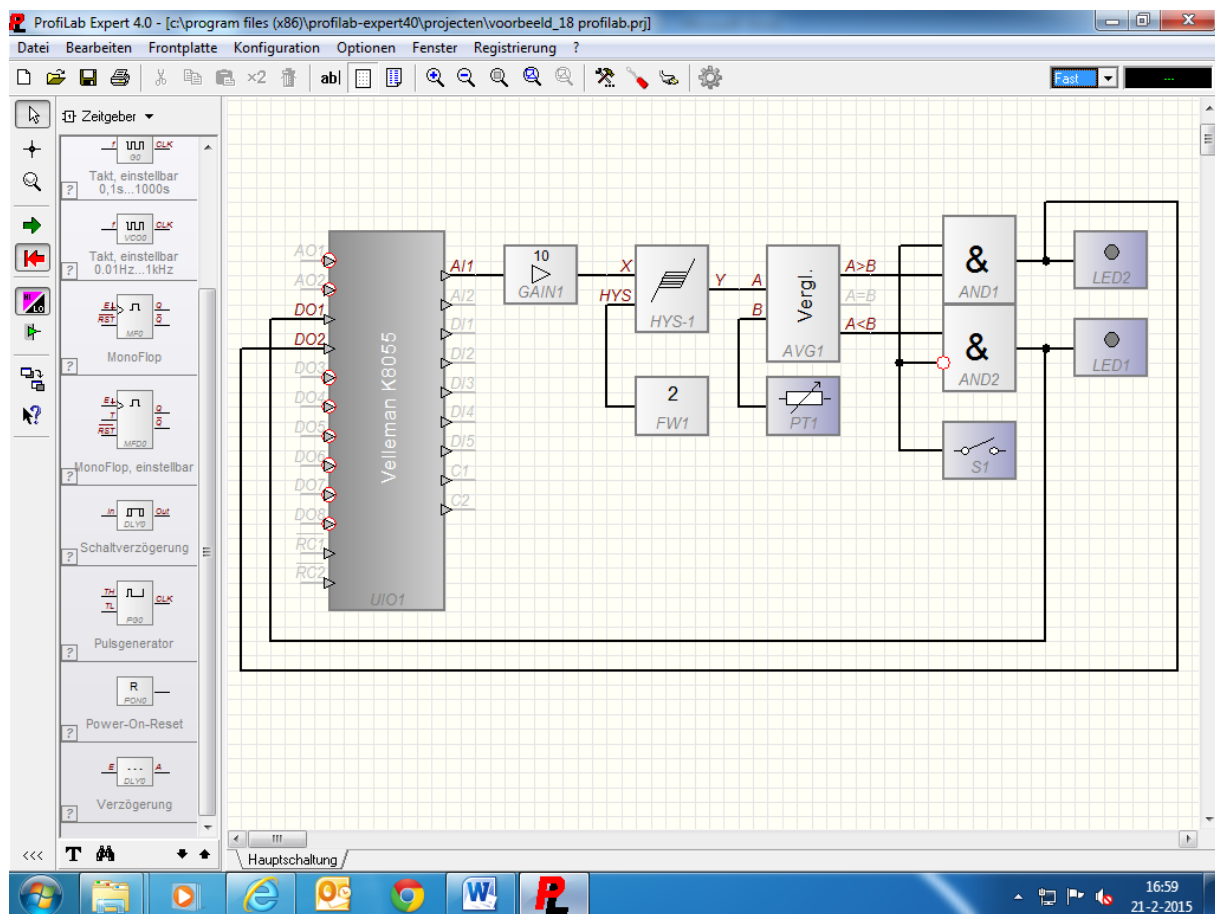


En zo zou je de frontplaat eruit kunnen laten zien.

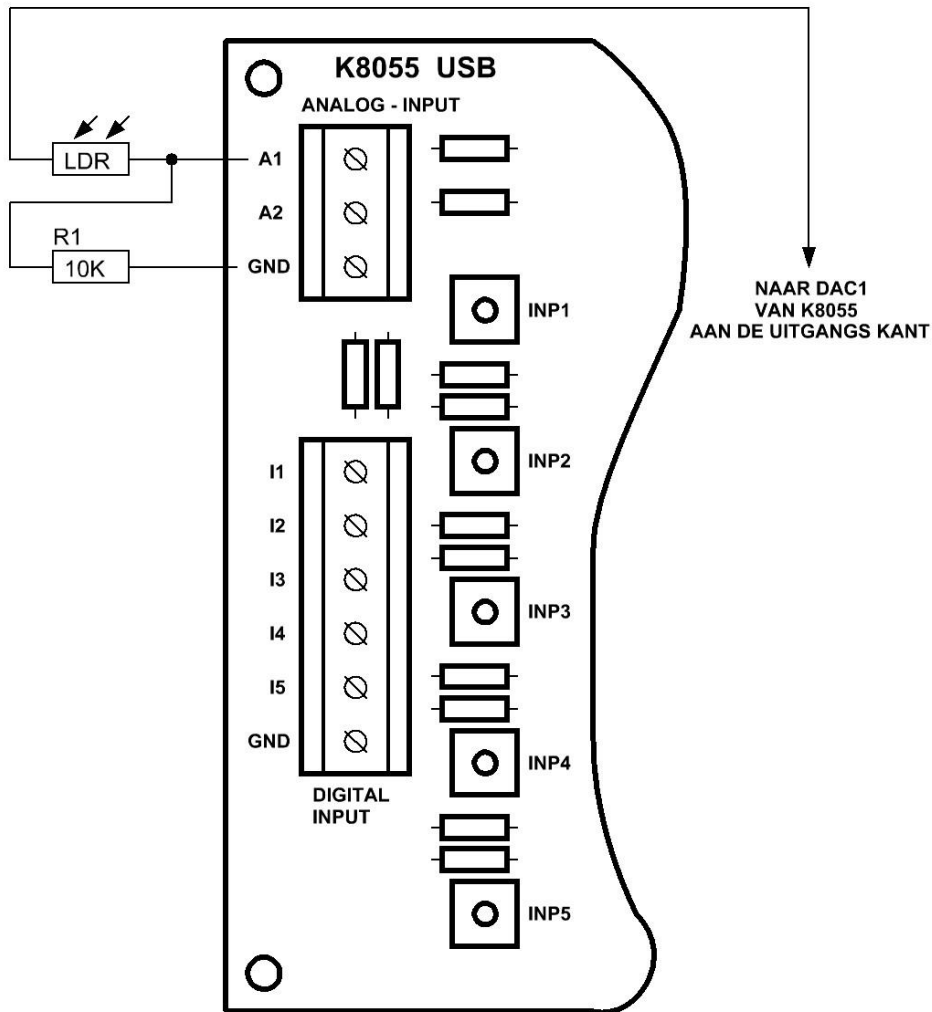


Bovenaan zie je de leds voor de stapvolgorde, en daar onder de run schakelaar en de keuze schakelaar voor links en rechtsom.

4.12: Licht donker schakeling



We gaan nu een licht donker regeling maken. We slepen één versterker module, één hysteresis module, één vaste waarde module, één vergelijker module, één potmeter, één schakelaar, twee and poorten, twee leds, en een K8055 module naar het werkblad. Dit gaan we weer met elkaar verbinden zoals in het voorbeeld te zien is. De analoge ingang AI1 van de K8055 is verbonden met de ingang van de versterker module GAIN1, de versterking van de module staat op 10. De uitgang van de versterker komt op ingang X van de hysteresis module HYS1, en de vaste waarde module FW1 komt op de HYS ingang van de hysteresis module. De waarde in de vaste waarde module FW1 staat op 2. De uitgang Y van de hysteresis module HYS1 komt op de ingang A van de vergelijking module AVG1, op ingang B van de module komt de potmeter PT1 te zitten. De uitgangen van vergelijker AVG1 $A < B$ en $A > B$ komen op de ingangen van de and poorten AND1 en AND2 te zitten, de andere twee ingangen komen aan de schakelaar S1 te zitten. Schakelaar S1 is geconfigureerd als keuze schakelaar, dit kan gedaan worden onder eigenschappen op de frontplaat. Zoals te zien is op het voorbeeld moet één van de ingangen van de and poort AND2 geïnverteerd worden, dit kan je doen door op de desbetreffende ingang te klikken. Als het goed is verschijnt er nu een rondje op de pin van de and poort AND2. De uitgangen van de and poorten AND1 en AND2 komen aan de leds te zitten, en ze zijn ook verbonden met DO1 en DO2 van de K8055. Op deze manier kan je ook nog iets schakelen als je dat zou willen.



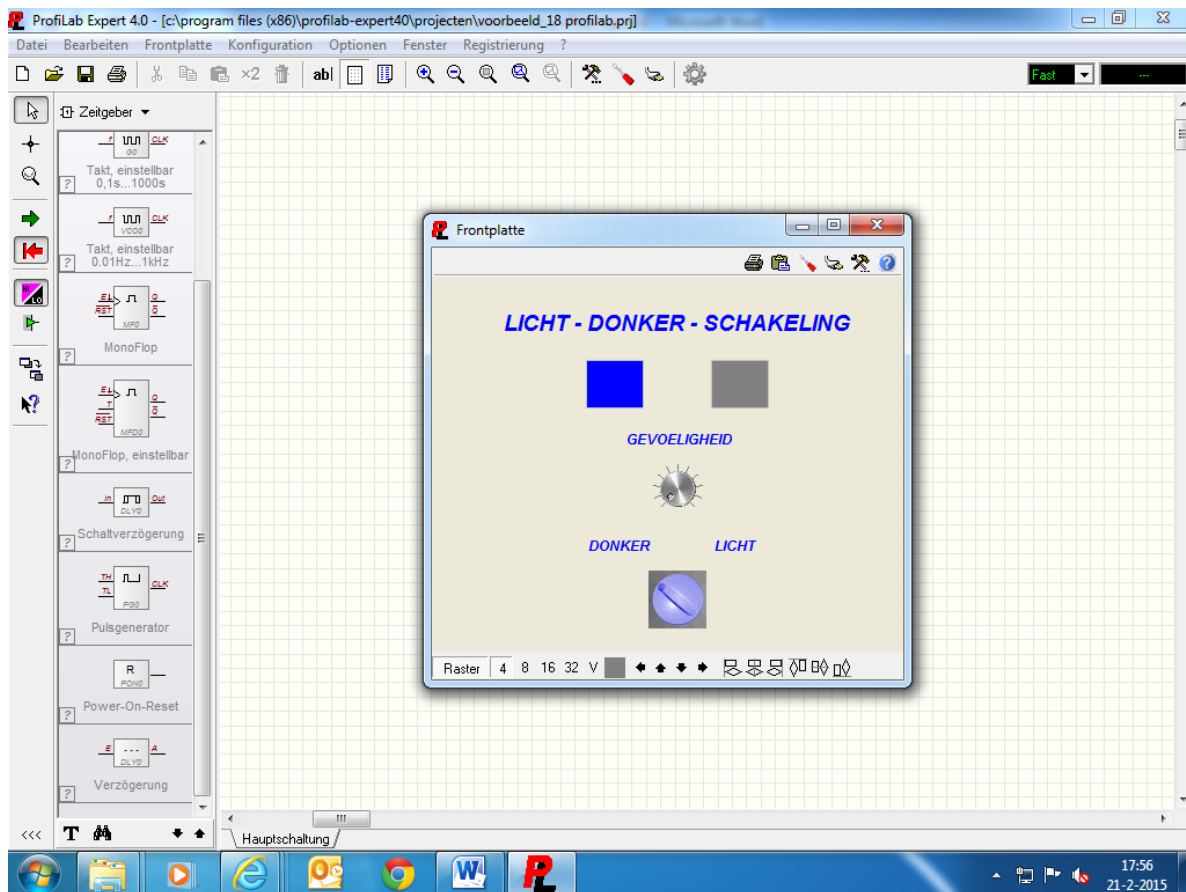
Op deze manier kan de LDR aan gesloten worden op de K8055. De LDR komt op de analoge ingang A1 van de K8055. Op het knooppunt van de ingang komt een 10K weerstand naar de GND aansluiting van de K8055. De andere kant van de LDR komt aan de uitgang kant op de DAC1 aansluiting te zitten, op de analoge uitgang dus. Die uitgang kan je via de jumper op de K8055 activeren zodat er 5VDC op de DAC1 pin kont te staan.

De spanning op de DAC1 pin kan je instellen met de desbetreffende potmeter die op de K8055 zit.

De instelling van de potmeter moet even proefondervindelijk gedaan worden, daar moet een beetje mee geëxperimenteerd worden.

De frontplaat komt er zo uit te zien als het goed is, of je maakt zelf weer een indeling.

Zie voorbeeld op Blz. 44.



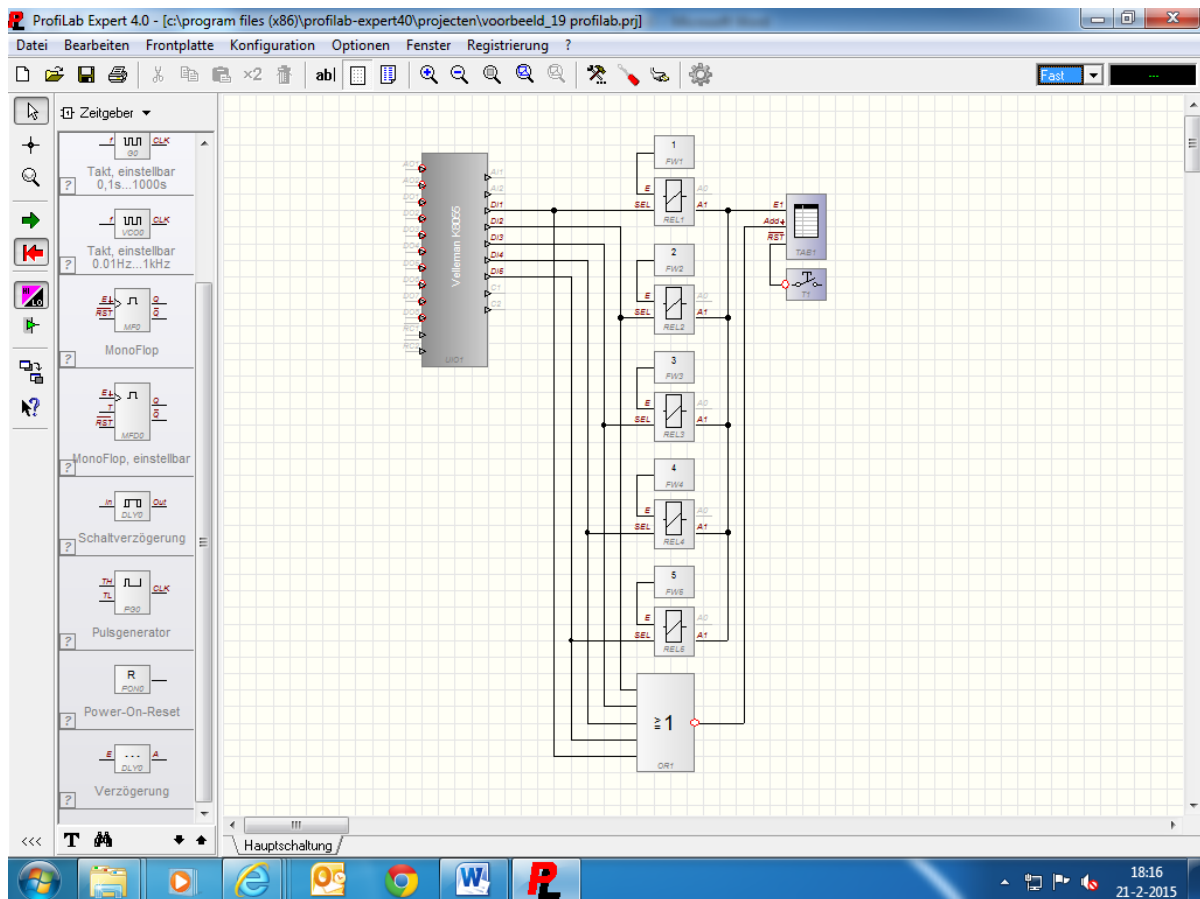
Met de potmeter op de frontplaat kan je de gevoeligheid instellen waarbij de leds oplichten op de frontplaat. Ook worden de uitgangen DO1 of DO2 geactiveerd. Met die uitgangen kan er bijv. een relais geschakeld worden, of bijv. een zoemer.

Met de keuze schakelaar S1 kan er op licht en donker geselecteerd worden.

4.13: Data opslaan

We gaan een schakeling maken waarmee je data op kan slaan. We slepen vijf vaste waarde modules, één or poort, vijf analoge uitgang relais, één tabel module, één schakelaar, en een K8055 module naar het werkblad. We sluiten de ingangen DI1, DI2, DI3, DI4, en DI5 van de K8055 aan op de select ingangen van de analoge relais REL1, REL2, REL3, REL4, en REL5. De E ingangen van de analoge relais sluiten we aan op de vaste waarde modules, de waarde van die modules stellen we in op 1, 2, 3, 4, en 5. De uitgangen van de analoge relais komen op de ingang E1 van de tabel module TAB1. De schakelaar T1 komt op de reset ingang RST van de tabel module. De schakelaar T1 wordt ook geïnverteerd. De ingangen van de K8055 lopen ook via de or poort naar de Add ingang van de tabel module. De uitgang van de or poort wordt weer geïnverteerd.

Zie voorbeeld op Blz. 45.

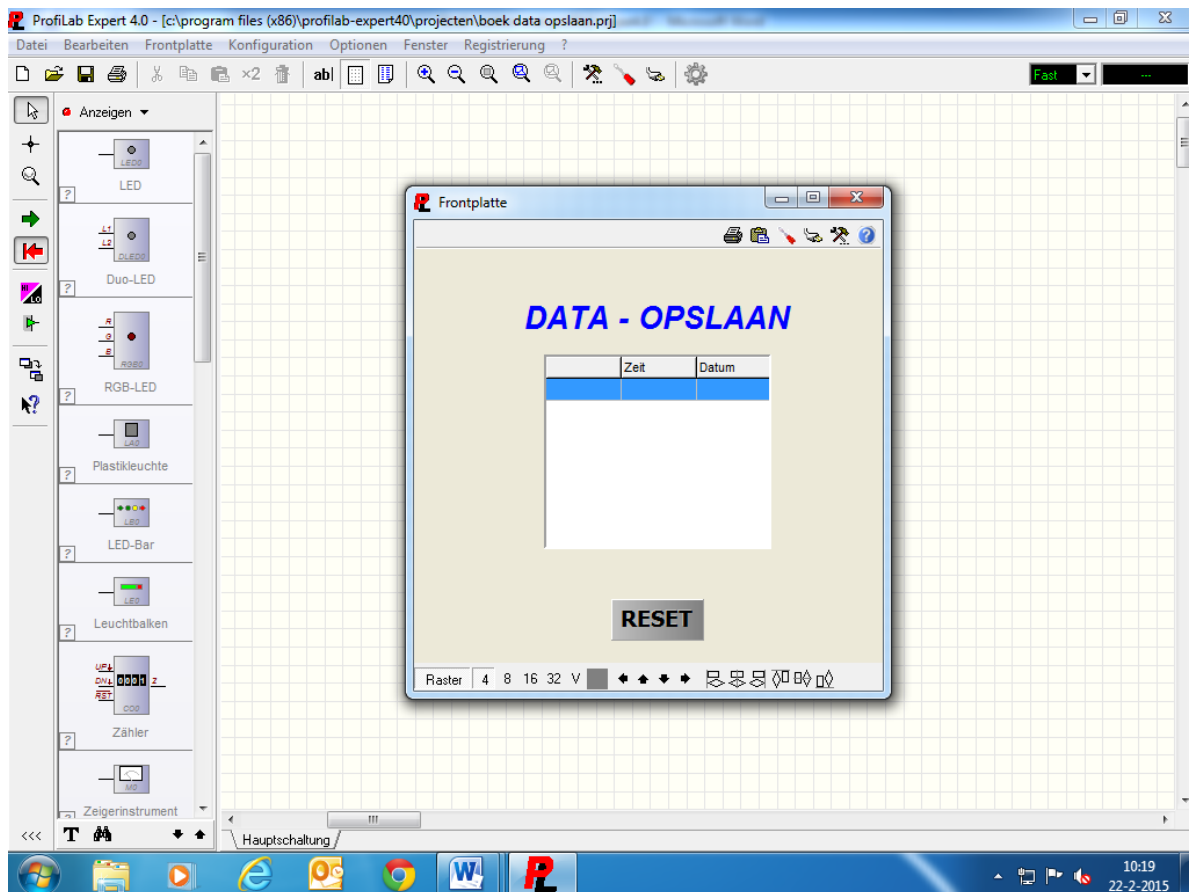


Als er nu op één van de knoppen van de K8055 gedrukt wordt, dan zal de waarde die daar aanhangt in de tabel geschreven worden. Als er bijv. op knop één gedrukt wordt dan komt het getal één in de tabel te staan. Het analoge relais geeft deze waarde door aan de ingang E1 van de tabel module TAB1. Als je bijv. knop vijf indrukt wordt de waarde vijf weg geschreven naar de tabel. Als er op T1 gedrukt wordt, dan wordt de inhoud uit de tabel gewist.

Ik heb de frontplaat op deze manier ingedeeld, maar dat kan weer naar eigen inzicht gedaan worden.

Zoals je ziet geeft de tabel ook de datum en de tijd aan, ook is te zien dat het eerste vak nog leeg is. Daar kan men zelf nog een tekst ingeven, als je naar de frontplaat gaat dan kan je daar onder eigenschappen nog dingen aanpassen. In het eerste vak zou je bijv. DATA, TEMP, WAARDE, ENZ, ENZ in kunnen vullen, net wat er nodig is. Ook kan men de tekst grote, de tekst kleur, of het letter type ingeven. Er is ook nog een mogelijkheid om de waarde naar een Word of Exel document te laten schrijven, op deze manier kan je bijv. in Excel een tabel maken waar de datum, tijd, en waarde in komt te staan. Maar over het opslaan in Excel of Word later meer. Zoals je ziet is er heel veel mogelijk, ik ga hier niet allemaal voorbeelden voor maken omdat het dan wel heel veel wordt over één onderwerp. Dit is een mooie uitdaging om zelf te onderzoeken en te proberen.

Zie voorbeeld op Blz. 46.



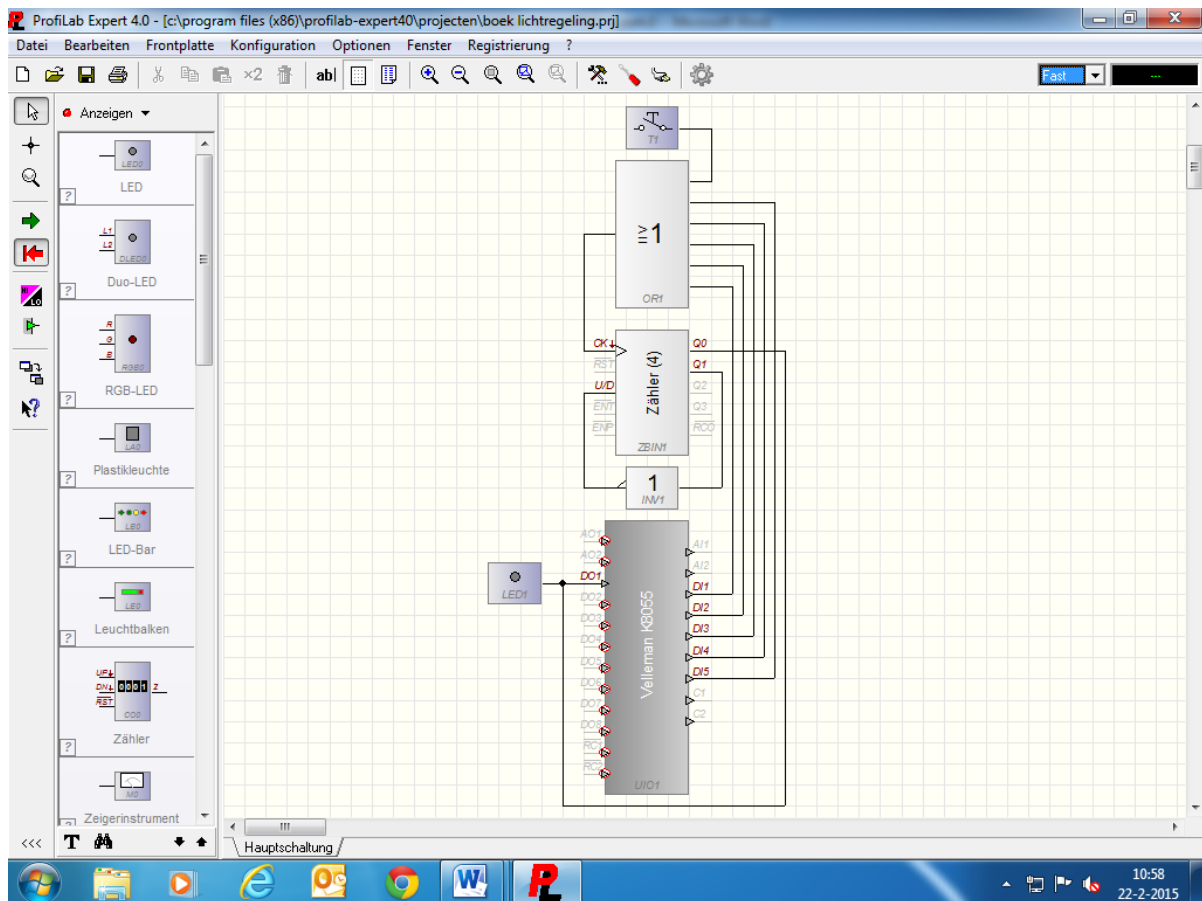
Met de Tabel kan eventueel ook een temperatuur logger gemaakt worden, dit kan bijv. via de analoge ingang gedaan worden, (zie voorbeeld licht donker schakeling) als de LDR vervangen wordt door bijv. een NTC dan kan je via een aangepaste formule de gemeten temperatuur wegschrijven in de tabel.

Met de reset knop op de frontplaat wordt de inhoud gewist, let er wel op dat als je de functie wegschrijven naar Word of Excel niet gebruikt, dat je alle data kwijt bent.

4.14: Licht effecten

Dit is een schakeling waarmee je verlichting op verschillende plaatsen aan en uit kan zetten. We slepen één drukknop, één led, één or poort, één teller module, één inverter, en een K8055 module naar het werkblad. De ingangen op de K8055 DI1, DI2, DI3, DI4, en DI5 worden verbonden met de or poort OR1, en aan de overgebleven ingang van OR1 komt de schakelaar T1. De uitgang van de or poort komt aan de CK ingang van teller ZBIN1. De uitgang Q0 van de teller ZBIN1 komt aan de uitgang DO1 van de K8055, en daar komt ook de LED1 aan te zitten. De uitgang Q1 van de teller ZBIN1 gaat via de inverter INV1 naar de ingang U/D van de teller ZBIN1.

De teller is op deze manier zo geschakeld dat als er op een knop wordt gedrukt de teller uitgang Q0 hoog wordt, en als er nogmaals gedrukt wordt dan wordt de teller uitgang Q0 weer laag. Je krijgt op deze manier dus een toggle functie. Zie voorbeeld op Blz. 47.



Op de uitgang DO1 van de K8055 kan bijv. weer een relais gezet worden, zodat je een zwaardere last kan schakelen. Op deze manier kan je dus op zes verschillende plaatsen het licht aan en uit zetten.

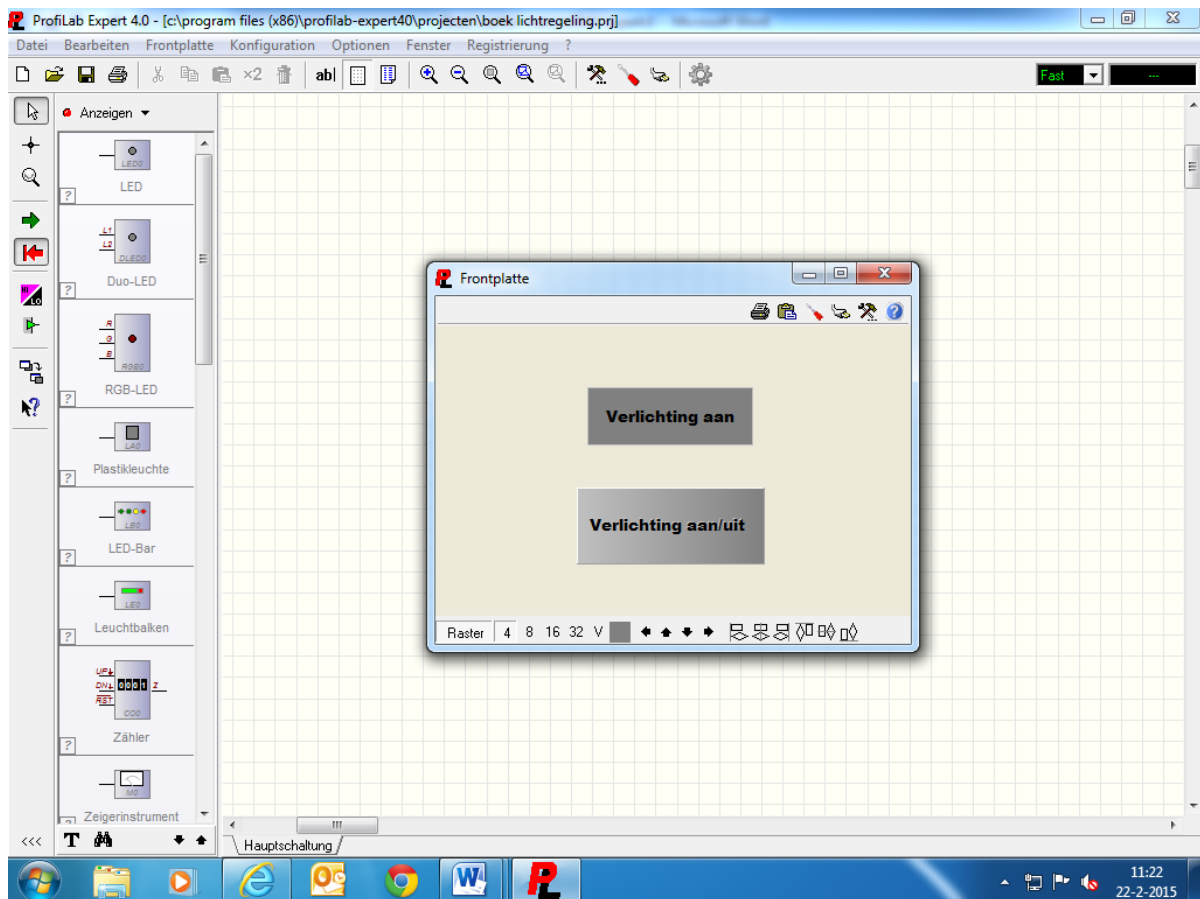
De frontplaat is zeer eenvoudig, er zit maar één drukknop op en één led.

Ook kan je via de drukknoppen die op de K8055 zitten de verlichting aan en uitschakelen.

Het is ook mogelijk om de schakelaars extern te plaatsen, dit kan bijv. op enkele meters van elkaar gebeuren. De kabels met de schakelaars worden dan aangesloten op de digitale ingangen DI1, DI2, DI3, DI4, en DI5 van de K8055, hoe dat gedaan kan worden is in het begin beschreven. Ook kan je het best een 100nF condensator over het schakel contact zetten, dit is voor het ont denderen van de schakelaars. Hoe het relais aangesloten kan worden kan je vinden onder het kopje digitale uitgangen.

Dit is een voorbeeld voor verlichting, maar het kan ook voor andere doeleinden gebruikt worden natuurlijk. Waar voor allemaal dat mag zelf ingevuld worden. Ook kan er gebruik gemaakt worden van meerdere teller modules, op deze manier kan je bijv. ook een relais bediening maken met bijv. acht relais. Dan krijg je dus acht puls schakelaars met een toggle functie. De mogelijk heden worden op deze manier eindeloos.

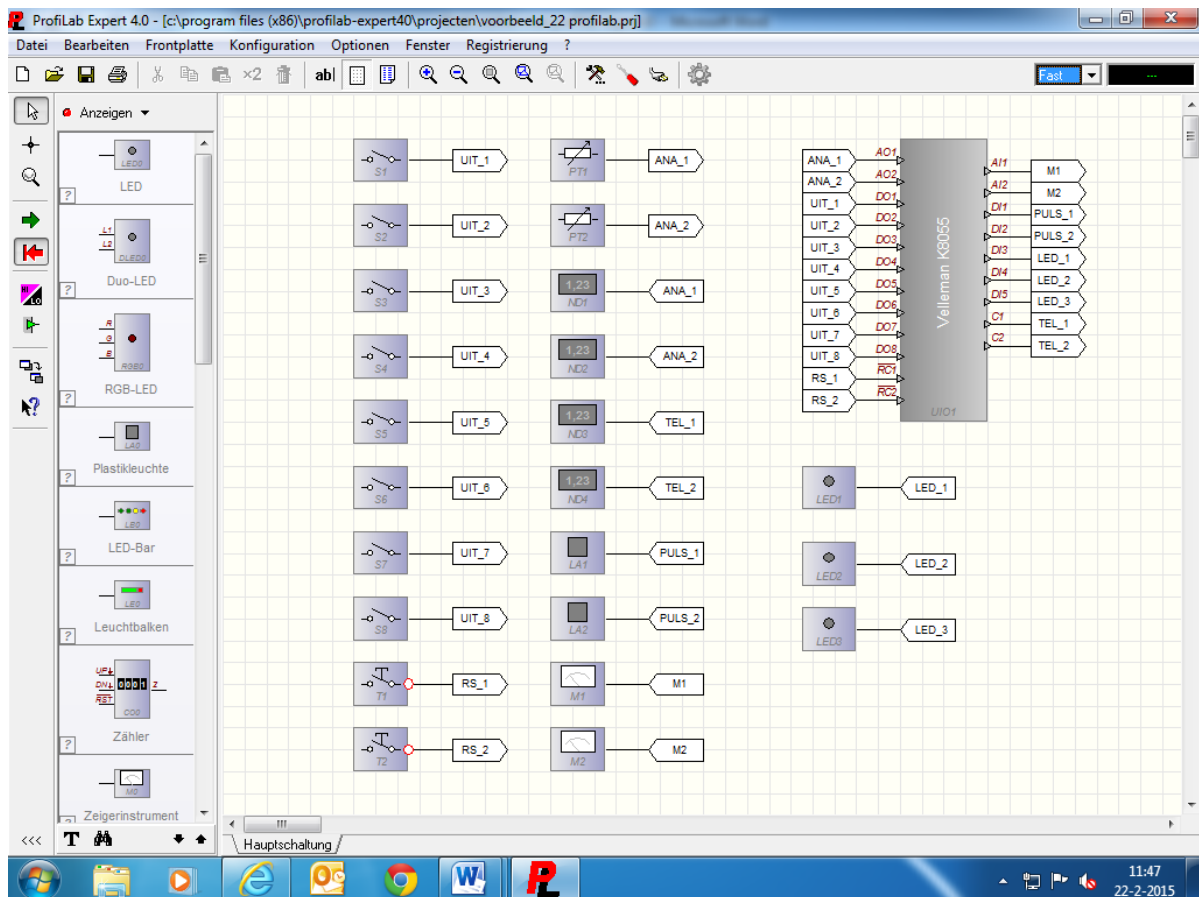
Zie voor frontplaat Blz. 48.



4.15: Programmeren met labels

We gaan nu een voorbeeld maken met labels. We slepen acht schakelaars, twee drukknoppen, twee potmeters, vier displays, twee lampen, twee analoge meters, drie leds, en een K8055 module naar het werkblad. De labels zijn te vinden onder diversen en dan selecteer je plus, massa, test. We zetten eerst alle componenten netjes onder en naast elkaar, (zoals op het werkblad te zien is) en daarna gaan we de labels plaatsen. We slepen eerst een label van het type SPRUNG naar het werkblad, als we daar dubbel op klikken krijg je een veld te zien waar je een waarde of een tekst in kan geven bijv. het label achter S1 geven we de naam UIT_1, bij S2 geven we de naam UIT_2 in. Dit doen we bij alle schakelaars S1 t/m S8, T1 en T2, en PT1 en PT2. Als we hier de namen voor aangemaakt hebben slepen we een label van het type SPRUNG (ZIEL) naar het werkblad. Als je daar nu dubbel op klikt en je klikt daarna op het pijltje naast het veld, dan krijg je de namen te zien die je aan de labels van het type SPRUNG gegeven hebt. Uit die lijst selecteer je bijv. de tekst UIT_1 en daarna druk je op OK, nu is het label SPRUNG UIT_1 en het label SPRUNG ZIEL UIT_1 met elkaar gekoppeld. Als je nu het label SPRUNG UIT_1 met S1 verbind, en label SPRUNG ZIEL UIT_1 met uitgang DO1 van de K8055 is de verbinding tot stand gekomen.

Zie voorbeeld op Blz. 49.



Dit doe je dus ook met de rest van de componenten, dus met de digitale ingangen, digitale uitgangen, analoge ingangen, analoge uitgangen, en de teller ingangen. Als dat gebeurt zoals het op het werkblad staat is alles gekoppeld aan de K8055.

De ingangen kunnen bediend worden met de knopjes op de K8055, maar er kunnen ook weer externe schakelaars gebruik worden. De ingangen DI1 en DI2 zijn voor de tellers die op de K8055 zitten, de waarde van de tellers wordt weer gegeven op de twee displays links met de tekst aantal erbij. Op de leds erboven is de tel puls te zien, met de drukknop eronder kan de teller gereset worden. De analoge uitgangen ANALOOG-UIT_1 en ANALOOG-UIT_2 kunnen met de potmeters op de frontplaat bediend worden, de analoge ingangen ANALOOG-IN_1 en ANALOOG-IN_2 worden op de wijzer meters weergegeven. Deze ingangen kunnen bediend worden door de potmeters die op de K8055 zitten, of door een spanning (0 T/M 5VDC) op de klemmen te zetten. De drie digitale ingangen IN_1, IN_2, en IN_3 zitten op de drukknoppen 3, 4, en 5 op de K8055. De acht digitale uitgangen UIT_1 T/M UIT_8 zijn de drukknoppen op de frontplaat, en daar worden de digitale uitgangen op de K8055 mee bediend.

De uitgangen kunnen weer van relais voorzien worden, en daar kan weer van alles mee geschakeld worden.

Voor frontplaat zie Blz. 50.

De indeling van de frontplaat heb ik zo gedaan.



Tot zover schakelingen rond de K8055.

Hoofdstuk 5: Over Pic controllers en interfaces

We gaan nu bespreken hoe we Profilab samen kunnen gebruiken met de controllers van Microchip. Dat zijn de Pic microcontrollers, dit is maar één merk van de vele controllers die er te koop zijn. Maar ik heb destijds voor deze controllers gekozen, omdat ik ze fijn vindt werken. Ook is er veel over te vinden, er is ook een site waar een beginners cursus beschreven staat. Hier kan je van alles vinden over het starten met micro controllers www.picbasic.nl onder het kopje PIC BASIC staan 10 lessen voor beginners. We gaan met deze controller de 16F887 wat voorbeelden maken, ik heb overal printen voor gemaakt maar het kan ook op een bread board of gaatjesprint opgezet worden natuurlijk.

5.1: Controllerboard met 16F887

Ik heb een controllerboard opgezet rond de 16F887, dit is een veertig pin controller. Het board is ook geschikt voor de 16F877, 16F877A, 16F887, en de 16F1939, deze controllers zijn allemaal pin compatible. De controller draait op een kristal van 10MHz, dat is snel genoeg voor alle toepassingen waar voor ik hem gebruikt heb.

Alle pinnen worden naar buiten gebracht op de print, bij iedere pin (in of uitgang) staat ook een plus en een massa. Dit is gedaan omdat je dan makkelijk sensoren en schakelaars aan kan sluiten op het board. Er zit ook een ICSP (in circuit serieel programmeer) aansluiting op, zodat je de controller tijdens het programmeren niet uit het board hoeft te halen.

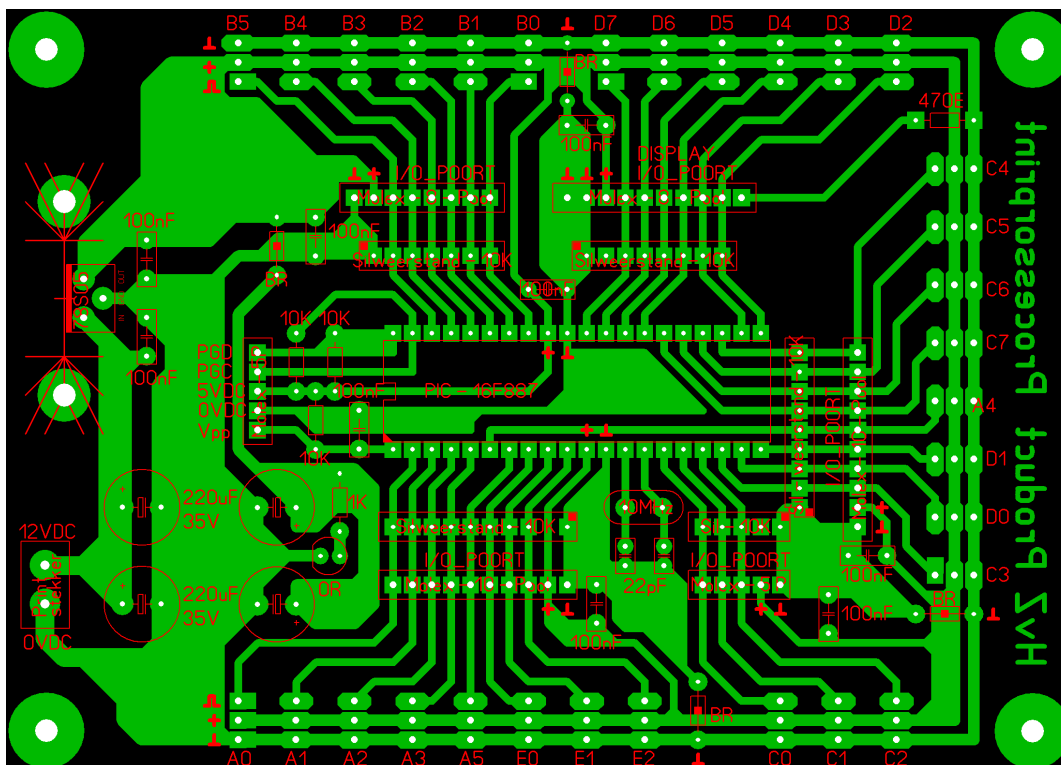
De connectoren zijn van het type Molex, drie, vijf, acht, en tien polig. Er is ook één connector speciaal voor het aansluiten van bijv. een display, het display kan daar direct op aangesloten worden zonder tussenkomst van een interface. Het display wordt aangesloten op de D poort.

Ook is er een connector speciaal voor bijv. een motorregelaar aanwezig, hier zitten de pwm pinnen van de controller op. Op de connector zitten twee pwm signalen en één vrijgave signaal. De connector voor het display en voor de motorregelaar kunnen ook als normale in en uitgang pinnen gebruikt worden natuurlijk.

Het board is ook voorzien van pull down weerstanden aan iedere pin, dit zijn 10K sil weerstanden. Dit type weerstand neemt weinig ruimte in beslag.

Het board heeft ook een eigen spanningsregelaar aan boord, dit is een 78S05. Deze regelaar voorziet het board van 5VDC dit wordt gebruikt voor de controller, maar ook om eventueel sensoren en schakelaars te voeden die aan het board hangen. Ook kan de spanning gebruikt worden om eventuele interfaces te voeden. De seriële interface voor de data overdracht naar de pc wordt ook gevoed uit de 5VDC die op het board zit. Maar daar over later meer. De voedingsspanning voor het board mag tussen de 9 en 24VDC zijn.

5.2: Print controllerboard



De print is getekend in SPRINT – LAYOUT, dat is ook een programma van ABACOM net als Profilab – expert 4.0. Het formaat van de print is 140 X 100mm.

5.3: Seriële interface met PC817

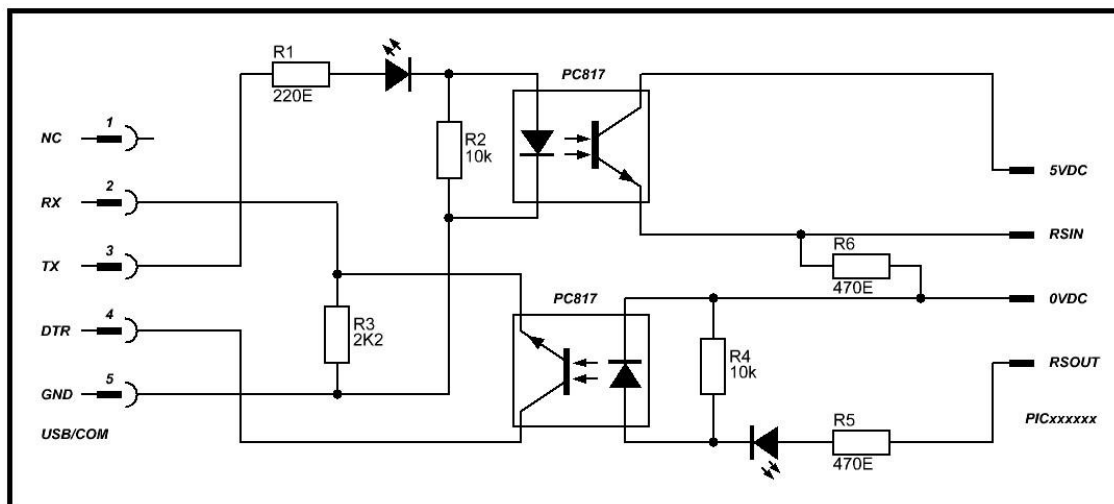
Om de 16F887 samen te laten werken met Profilab is er een interface nodig, de interface zorgt ervoor dat er data uitgewisseld kan worden tussen de pc en de controller. De interface zorgt ervoor dat de verschillende spanning niveaus niet voor problemen kan zorgen. De spanning op de seriële poort van een pc kan variëren tussen de +/- 6 en -6, en 12 en -12VDC. En aangezien het spanning niveau van de controller 5VDC is gaat dat zonder interface niet werken. Normaal wordt zoiets met een MAX - RS232 interface gedaan, maar ik heb gekozen voor een paar optocouplers van het type PC817. En ik moet zeggen dat het perfect werkt, plus dat je gelijk een galvanische scheiding hebt tussen de twee signalen.

De interface haalt met gemak 9600 baud, en dat is snel genoeg voor de toepassing waar ik het voor gebruik.

De interface kan ook samen met een usb naar serieel converter gebruikt worden. Zelf gebruik ik daar een Sabrent converter voor. Op deze manier is het mogelijk om te communiceren via de usb poort. Veel laptops hebben tegenwoordig geen seriële poort meer, dus dan is dit een mooie oplossing.

In alle voorbeelden wordt gebruik gemaakt van de Sabrent usb serieel converter, dus als ik het over een seriële com poort heb zijn het usb poorten. Ik ga gebruik maken van COM5 in de voorbeelden.

5.4: Schema seriële interface



Zoals op het schema te zien is hebben we niet veel onderdelen nodig voor de interface. Aan de twee leds is te zien of er communicatie is tussen de pc en de controller. De aansluiting aan de controller kant wordt weer gedaan met een vier polige Molex connector. Aan de kant van de pc wordt gebruik gemaakt van een negen polige haakse SUB – D connector. Er worden maar vier aansluitingen gebruikt van de connector, RX, TX, DIR, en de GND.

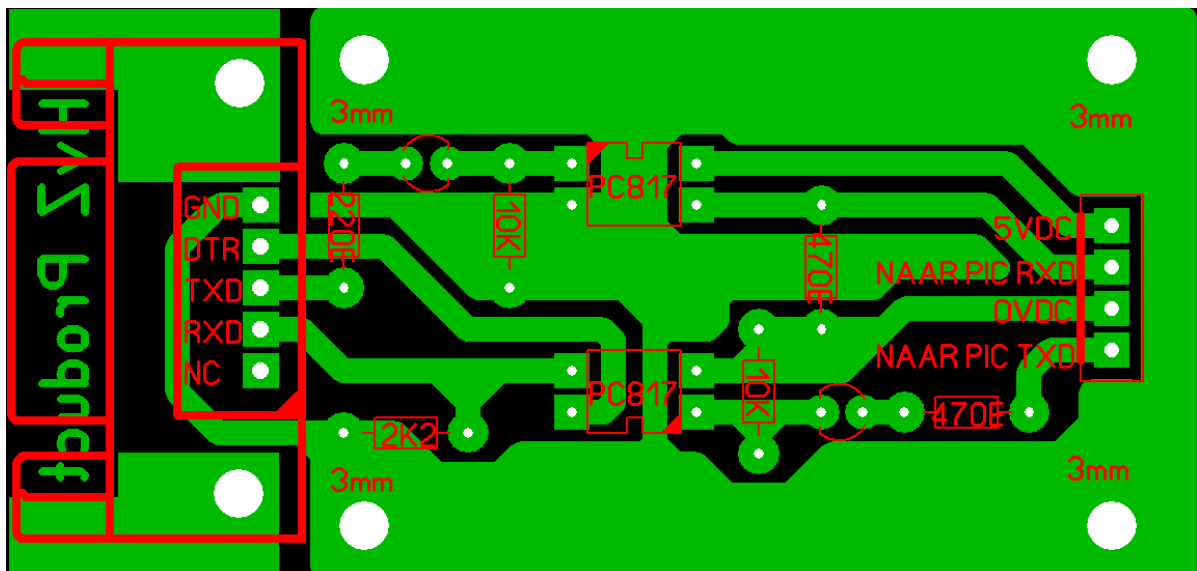
Als Profilab wordt opgestart op de pc dan wordt de DTR pin hoog, de spanning die daar dan op komt kan variëren tussen de 6 en 12VDC. Die spanning wordt door de onderste PC817 doorgestuurd naar de RX pin van de seriële poort van de pc. Dit gebeurt alleen als er een signaal op de RSOUT pin komt te staan dat afkomstig is uit de controller.

Als er een signaal uit de pc verzonden moet worden naar de controller, dan komt er een signaal op de TX pin te staan en dat signaal wordt dan door de bovenste PC817 doorgezonden naar de controller. Zoals je ziet hangt één kant van de PC817 aan de 5VDC, die spanning is afkomstig van de controller. Dat signaal komt dan op de RSIN pin van de controller binnen.

Op deze manier wordt er dus gecommuniceerd tussen de pc en de controller, en op deze manier blijven de twee signaal niveaus mooi gescheiden van elkaar.

Hier is ook weer een printje voor gemaakt.

5.5: Print seriële interface



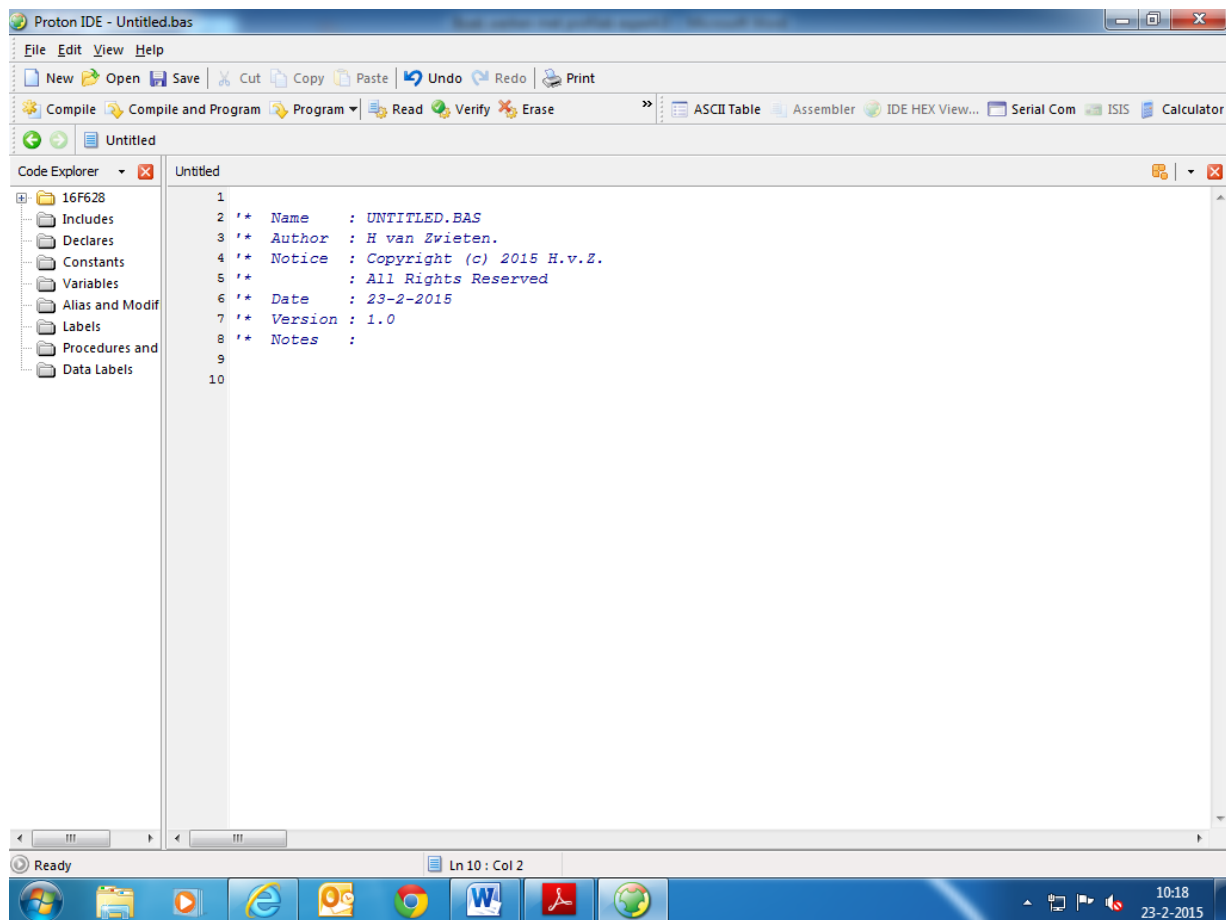
Het is een enkelzijdige print met een maat van 73 X 35mm. Dit kan natuurlijk ook weer op gaatjesprint gemaakt worden.

5.6: Over het programma Proton

Proton is een programma waar in de code voor de controller geschreven wordt, de taal waarin geschreven wordt heet pic – basic. Maar het programma voor de controller kan ook in C, C++, assembler, en bijv. jall geschreven worden. Maar ik heb destijds voor Basic gekozen en daar gebruiken we Proton voor. Het programma bevat ook een compiler, met een compiler wordt de geschreven code omgezet naar machine taal. Machine taal is eigenlijk niet meer dan een hexadecimaal code waarmee iedere controller werkt.

Waar ik zelf mee werk is een gekocht programma, en daar hangt dus een prijs kaartje aan. Er kan ook met een gratis versie gewerkt worden, maar dan kan je maar tot vijftig regels programmeren. Dat is op zich niet zo veel natuurlijk, maar het is wel te doen voor kleine stukjes code. Ook worden lang niet alle controller types ondersteund, en dat is met de gekochte versie wel zo. Maar het is een keuze die men zelf moet maken. De gratis versie is te downloaden bij www.picbasic.nl Voor de gekochte versie kan men hier terecht, www.protonbasic.co.uk en daar is nog veel meer te vinden. Er is bijv. ook een forum waar je allerlei vragen kan stellen over het werken met Proton, en je krijgt er ook ondersteuning als er eventueel problemen zijn. En je kan er natuurlijk je programma gratis upgraden als er nieuwe ontwikkelingen zijn rond Proton.

Hier onder heb je het werkblad van het programma, zoals je kan zien kan je daar je eigen programma kopje aanmaken met je eigen naam erin.



The screenshot shows the Proton IDE interface. The main window is titled 'Proton IDE - Untitled.bas'. The menu bar includes 'File', 'Edit', 'View', and 'Help'. The toolbar contains icons for 'New', 'Open', 'Save', 'Cut', 'Copy', 'Paste', 'Undo', 'Redo', and 'Print'. Below the toolbar are buttons for 'Compile', 'Compile and Program', 'Program', 'Read', 'Verify', and 'Erase'. On the right side of the toolbar, there are icons for 'ASCII Table', 'Assembler', 'IDE HEX View...', 'Serial Com', 'ISIS', and 'Calculator'. The 'Code Explorer' on the left shows a tree view with folders for 'Includes', 'Declares', 'Constants', 'Variables', 'Alias and Modif', 'Labels', 'Procedures and', and 'Data Labels'. The main editor area shows the following code:

```
1
2 '* Name : UNTITLED.BAS
3 '* Author : H van Zwieten.
4 '* Notice : Copyright (c) 2015 H.v.Z.
5 '* : All Rights Reserved
6 '* Date : 23-2-2015
7 '* Version : 1.0
8 '* Notes :
9
10
```

The status bar at the bottom shows 'Ready', 'Ln 10 : Col 2', and the system tray with the time '10:18' and date '23-2-2015'.

Dit kan je aanpassen door op de knop VIEW te klikken en daarna op de knop EDITOR OPTIONS en daarna op de knop PROGRAM HEADER te klikken. Daar kan je het naar eigen inzicht aanpassen. In dat zelfde scherm staat ook de knop GENERAL, daar kan je de indeling van het werkblad instellen. Onder de knop HIGHLIGHTER kan je een letter type kiezen, hoofdletters, onderstrepen, en de kleur van bepaalde teksten. De tekst achter '* NAME : kan altijd aangepast worden, daar vul je bijv. de naam in van je project. Het zelfde geldt bij '* NOTES : daar kan je bijv. nog een verklarende tekst in vullen wat met je project te maken heeft.

Onder de knop VIEW staat ook de knop CODE EXPLORER, als je daar op drukt krijg je links in beeld een veld te zien waar het type CONTROLLER te zien is waar op dat moment mee gewerkt wordt, en de INCLUDES files, DECLARES, CONSTANTS, VARIABLES, ALLAS, LABELS, PROCEDURES, en de DATALABELS.

Voor de rest van het programma is er onder het knopje HELP, DOCUMENTS, PROTON COMPILER MANUAL nog heel veel te vinden over het programma. Ook staan er overal stukjes voorbeeld code in de help file. Als je de help file PROTON COMPILER MANUAL geopend hebt, staat er links een hele index waar je een onderwerp kan selecteren.

Picbasic werkt met een aantal commando's, hier onder een opsomming van alle gebruikte commando's voor het werken met basic.

@, ADIN, ASM, ENDASM, BOX, BRANCH, BREAK, BRESTART, BSTART, BSTOP, BUSACK, BUSIN, BUSOUT, BUTTON, CALL, CDATA, CF_INIT, CF_SECTOR, CF_READ, CF_WRITE, CIRCLE, CLEAR, CLEARBIT, CLS, CONFIG, CONFIG1, CONFIG2, CONTEXT RESTORE, CONTEXT SAVE, CONTINUE, COUNTER, CREAD, CREAD8, CREAD16, CREAD32, CURSOR, CWRITE, DATA, DEC, DECLARE, DELAYCS, DELAYMS, DELAYUS, DEVICE, DIG, DIM, DISABLE, DTMFOUT, EDATA, ENABLE, END, EREAD, EWRITE, FOR, TO, NEXT, STEP, FREQOUT, GETBIT, GOSUB, GOTO, HBRESTART, HBSTART, HBSTOP, HBUSACK, HBUSOUT, HIGH, HPWM, HRSIN, HRSOUT, HRSIN2, HRSOUT2, HSERIN, HSEROUT, HSERIN2, HSEROUT2, I2CIN, I2COUT, IF, THEN, ELSEIF, ELSE, ENDIF, INC, INCLUDE, INKEY, INPUT, LCDREAD, LCDWRITE, LDATA, LEFT\$, LEN, LINE, LINETO, LOADBIT, LOOKDOWN, LOOKDOWNNL, LOOKUP, LOOKUPL, LOW, LREAD, LREAD8, LREAD16, LREAD32, MID\$, ON GOSUB, ON GOTO, ON GOTOL, ON_HARDWARE_INTERRUPT, ON INTERRUPT, ON_INTERRUPT, ON_LOW_INTERRUPT, OREAD, ORG, OUTPUT, OWRITE, PAUSE, PEEK, PIXEL, POKE, POP, POT, PRINT, PULSIN, PULSOUT, PUSH, PWM, RANDOM, RC5IN, RCIN, READ, REM, REPEAT, UNTIL, RESTORE, RESUME, RETURN, RIGHT\$, RSIN, RSOUT, SEED, SELECT, CASE, ENDSELECT, SERIN, SEROUT, SERVO, SET, SET_OSCAL, SETBIT, SHIN, SHOUT, SLEEP, SNOOZE, SONYIN, SOUND, SOUND2, STOP, STRN, STR\$, SWAP, SYMBOL, TOGGLE, TOWER, TOUPPER, TOSHIBA_COMMAND, TOSHIBA_UDG, UNPLOT, USBINIT, USBIN, USBOUT, USBPOLL, VAL, VARPTR, WHILE, WEND, XIN, XOUT.

Zoals men ziet zijn dat nog al wat commando's, maar vrees niet zoveel gaan we er niet gebruiken.

De commando's waar we mee gaan werken worden beperkt tot deze.

ADIN, ASM, ENDASM, CLEAR, CLS, CONFIG, DEC, DELAYMS, DECLARE, DIM, END, FOR, TO, NEXT, STEP, GOSUB, GOTO, SYMBOL, HPWM, TOGGLE, IF, THEN, ELSE, ENDIF, PRINT, RSIN, RSOUT, SERVO.

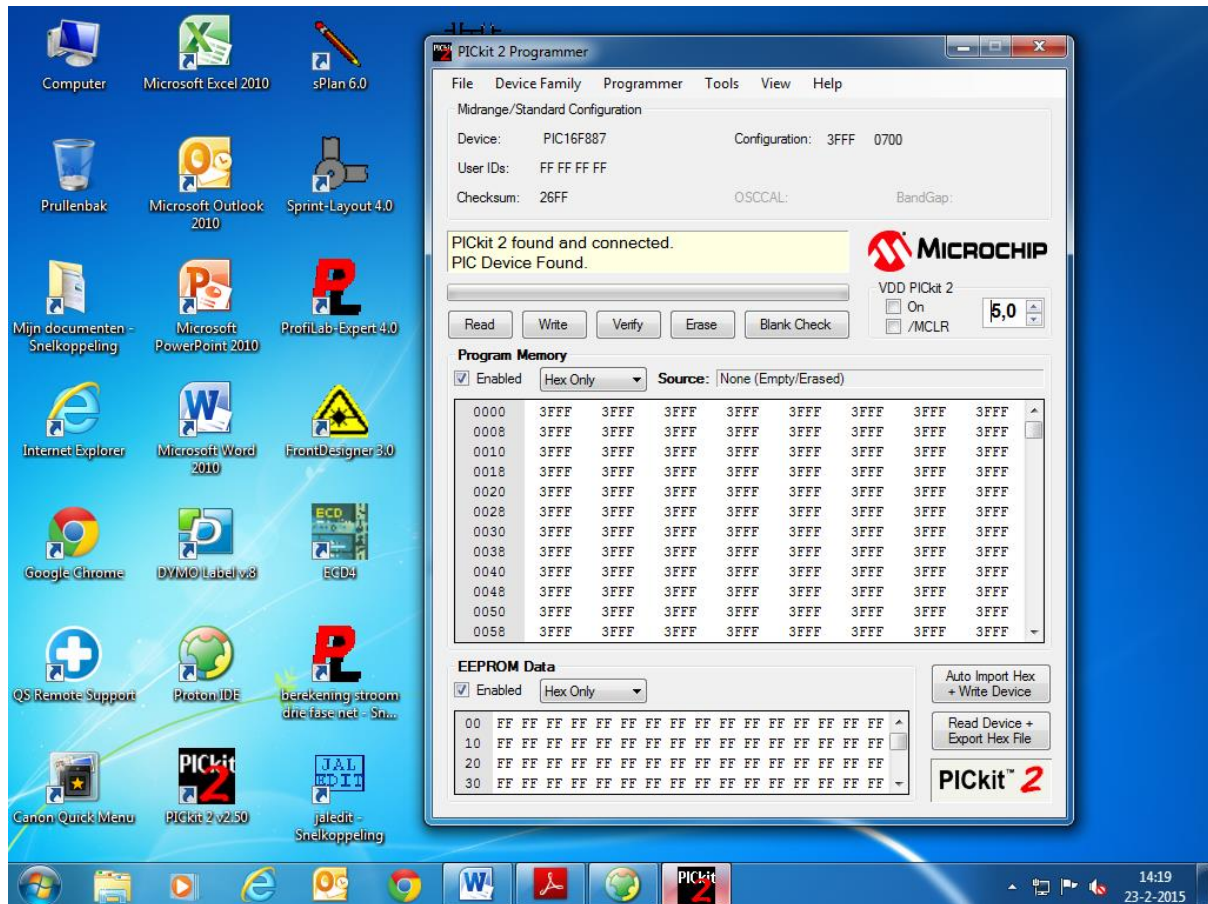
Dus dat valt nog wel mee misschien ben ik er een paar vergeten, maar die komen we dan vanzelf wel tegen in de voorbeelden. Maar daarover later meer.

Tot zover over het programma, later zien we hoe we het gaan gebruiken aan de hand van voorbeelden.

Alle basic software van de voorbeelden is getest, dus het is allemaal werkend.

5.7: Over de pickit_2

De pickit2, (van microchip) is een programmeer apparaat die de gecompileerde code in de controller zet. Dit gaat via de usb poort van een pc of laptop. De pickit2 wordt op de controller aangesloten via de ICSP connector die op het controller board zit. Zodra je de pickit2 aansluit op de pc of laptop en de controller, krijg je dit veld te zien.



Zoals te zien is op het veld wordt het type van de controller gelijk weer gegeven PIC16F887, ook staat er dat de pickit2 gevonden en verbonden is met de pc. Onder het knopje DEVICE FAMILY kan je ook handmatig een controller selecteren.

Onder het knopje PROGRAMMER kan je een selectie maken voor READ DEVICE, WRITE DEVICE, VERIFY, ERASE, BLANK CHECK, VERIFY ON WRITE, HOLD DEVICE IN RESET, WRITE ON PICKIT BUTTON, MANUAL DEVICE SELECT, PICKIT2 PROGRAMMER – TO – GO.

Onder het knopje TOOL kan je een selectie maken voor ENABLE CODE PROTECT, ENABLE DATA PROTECT, TARGET VDD & SET UNIT, USE VDD FIRST PROGRAM ENTRY, FAST PROGRAMMING, UART TOOL, CHECK COMMUNICATION, TROUBLE SHOOT.

Onder het knopje VIEW kan je een selectie maken voor SINGLE WINDOW, MULTI WINDOW. Onder het knopje FILE kan je een gecompileerde hex file naar de pickit2 halen, maar ook een gelezen hex file exporteren.

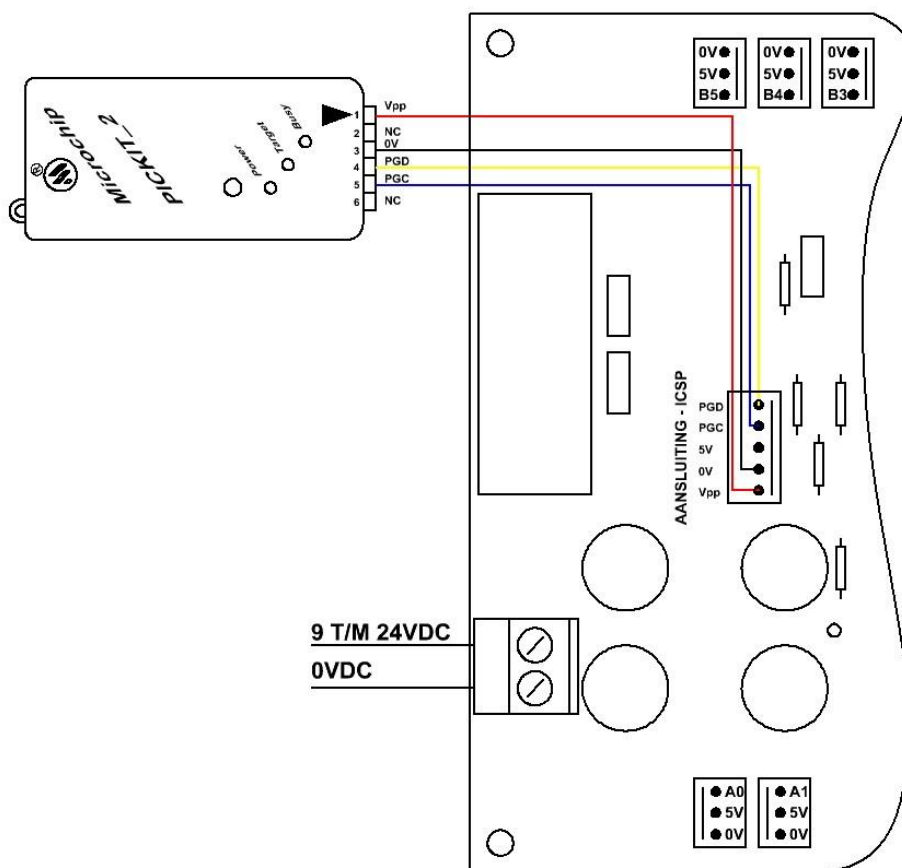
Op het veld er onder waar nu overal 3FFF staat, komt normaal het gecompileerde en geïmporteerde programma te staan. De hexadecimale code dus, op het veld daar onder komt de eventuele EEPROM DATA te staan.

Met de knoppen READ, WRITE, VERIFY, ERASE, en BLANK CHECK, kan je respectievelijk de controller lezen, schrijven, verifiëren, wissen, en controleren of de controller leeg is.

Als je een vinkje bij ON zet kan je de controller voeden van uit de pickit2, zelf gebruik ik dat niet omdat de controller een eigen voeding heeft. Met een vinkje bij MCLR kan je de controller resetten.

Dan kan je nog de voedingspanning instellen op de pickit2, deze zet ik zelf altijd op 5VDC.

Hier onder een schema hoe je de pickit2 aansluit op het controllerboard.



Pin één op de pickit2 is de VPP aansluiting, daar komt de programmeer spanning op voor de controller tijdens het programmeren. Pin twee is de 5VDC aansluiting. Pin drie is de 0VDC aansluiting. Pin vier is de PGD aansluiting, daar komt de data op tijdens het programmeren. Pin vijf is de PGC aansluiting, daar komt het klok signaal op tijdens het programmeren. En pin zes wordt niet gebruikt.

De 5VDC aansluiting **Pin twee** gebruik ik zelf ook niet, omdat ik de controller altijd apart voed met een externe voeding adapter. De pickit2 kan tijdens het testen van bijv. een programma altijd aangesloten blijven.

Hoofdstuk 6: Werken met Profilab en Pic controllers

Het is dus mogelijk om via Profilab samen te werken met controllers, je kan in Profilab een programma maken dat via de seriële poort of usb poort van een pc of laptop kan communiceren met de controller. Je hebt dan wel twee programma's nodig, één programma dat gemaakt is in Profilab en één programma dat geschreven is in basic (of een andere taal) voor de controller.

In Profilab kan je onder het knopje HARDWARE RS232 selecteren, als je dat gedaan hebt krijg je links een veld te zien waar je uit allerlei seriële modules kan kiezen.

Er kan allerlei data verzonden worden op deze manier, je kan bijv. data versturen van Profilab naar de controller en je kan data vanuit de controller naar Profilab verzenden. Dit kunnen zowel digitale als analoge signalen zijn.

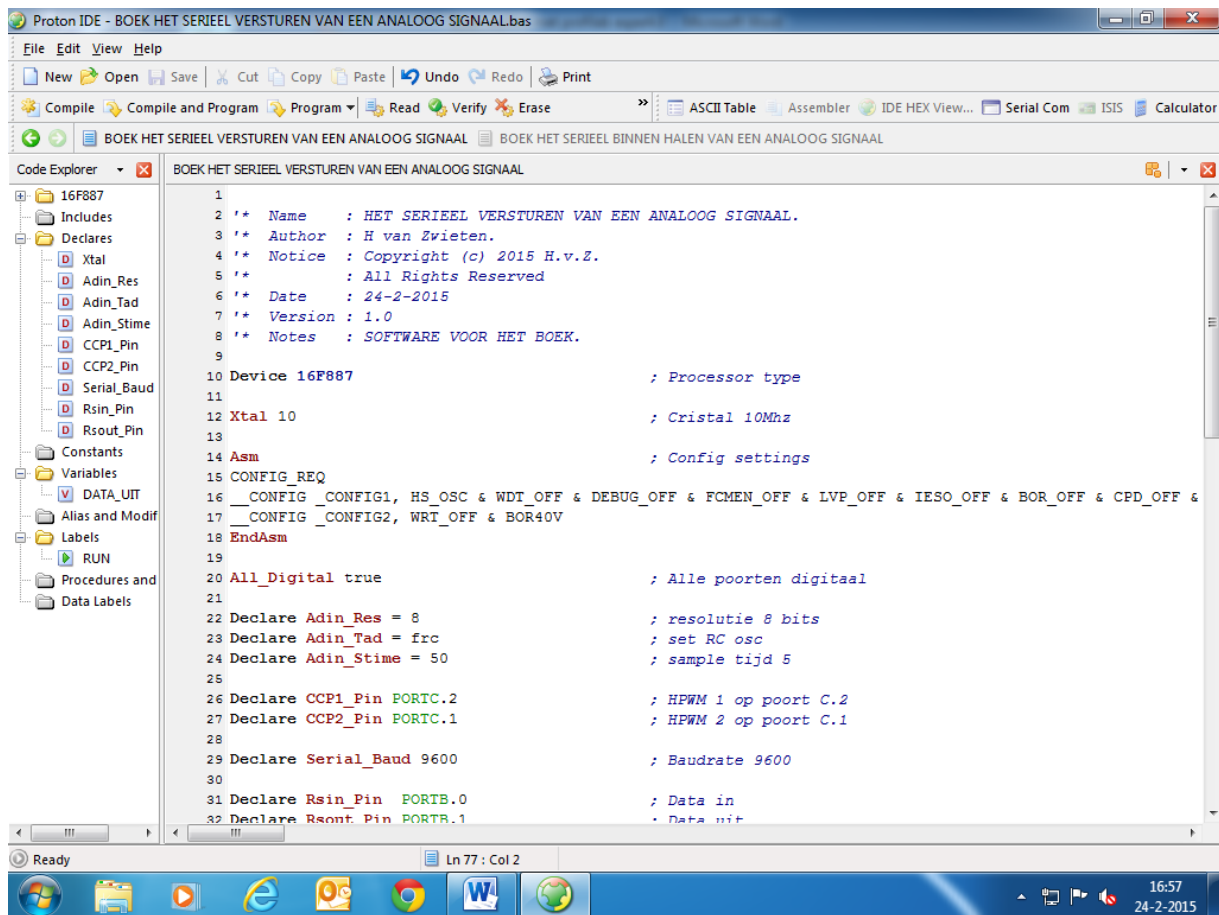
Je kan het een beetje vergelijken met de K8055, daar stuur je via de pc ook commando's naar toe. Alleen heb je daar het probleem dat je niet echt vrij bent in het kiezen van bijv. de pinnen die op de controller van de K8055 zitten. Omdat de pinnen allemaal benoemd zijn in het programma dat in de pic 16C745 zit, de 16C745 is het type pic dat gebruikt wordt op de K8055.

Wij gaan nu gebruik maken van de 16F887, en daar zijn we helemaal vrij in de keuze van het kiezen en indelen van de pinnen. Plus dat dit een veertig pin controller is, en de 16C745 maar een achtentwintig pin controller. Dus we hebben veel meer mogelijk heden op deze manier.

De mogelijkheden zijn eigenlijk onbegrensd, je kan er relais mee besturen, allerlei sensoren mee inlezen, motor besturingen maken, domotica projecten maken, enz., enz.

6.1: Het serieel binnen halen van een analog signaal

We gaan beginnen met het binnen halen van een analog signaal dat verzonden wordt door de controller. We gaan een potmeter inlezen met de controller, en dit serieel verzenden naar Profilab. We gaan hier PORTA.0 voorgebruiken van de controller, daar kan je een analoge ingang van maken. Dat kan je selecteren in de software, maar daar over straks meer. Voor de seriële overdracht gaan we de pinnen van PORTB.0 en PORTB.1 gebruiken. PORTB.0 wordt RSIN, en PORTB.1 wordt RSOUT. Het RSIN en RSOUT signaal gaat naar de interface, (die we al besproken hebben onder 5.3) en de uitgang van de interface TX en RX gaat naar de pc. We gaan nu eerst het programma maken voor de controller, en daarna het programma voor Profilab.



Hierboven staat het eerste programma geschreven in basic, het is alleen niet helemaal te zien. We gaan het programma hier onder helemaal plaatsen.

```

'* Name : HET SERIEEL VERSTUREN VAN EEN ANALOOG SIGNAAL.
'* Author : H van Zwieten.
'* Notice : Copyright (c) 2015 H.v.Z.
'* : All Rights Reserved
'* Date : 24-2-2015
'* Version : 1.0
'* Notes : SOFTWARE VOOR HET BOEK.

Device 16F887 ; Processor type

Xtal 10 ; Cristal 10Mhz

Asm ; Config settings
CONFIG_REQ
__CONFIG __CONFIG1, HS_OSC & WDT_OFF & DEBUG_OFF & FCMEN_OFF & LVP_OFF & IESO_OFF & BOR_OFF & CPD_OFF & CP_OFF & MCLRE_OFF & PWRTE_ON
__CONFIG __CONFIG2, WRT_OFF & BOR40V
EndAsm

All_Digital true ; Alle poorten digitaal

Declare Adin_Res = 8 ; resolutie 8 bits
Declare Adin_Tad = frc ; set RC osc
Declare Adin_Stime = 50 ; sample tijd

```

```

Declare CCP1_Pin PORTC.2           ; HPWM 1 op poort C.2
Declare CCP2_Pin PORTC.1           ; HPWM 2 op poort C.1

Declare Serial_Baud 9600           ; Baudrate 9600

Declare Rsin_Pin PORTB.0           ; Data in
Declare Rsout_Pin PORTB.1         ; Data uit

Dim DATA_UIT As Byte           ; Data serieelepoort compoort

Clear                               ; Wis geheugen

                                ;543210           ; Hulpregel poort A
PORTA = %000000                   ; Maak poort A laag
TRISA = %111111                   ; Poort_A I/O

                                ;543210           ; Hulpregel poort B
PORTB = %000000                   ; Maak poort B laag
TRISB = %000001                   ; Poort_B I/O

                                ;76543210          ; Hulpregel poort C
PORTC = %00000000                 ; Maak poort C laag
TRISC = %00000000                 ; Poort_C I/O

                                ;76543210          ; Hulpregel poort D
PORTD = %00000000                 ; Maak poort D laag
TRISD = %00000000                 ; Poort_D I/O

                                ;210              ; Hulpregel poort E
PORTE = %000                       ; Maak poort E laag
TRISE = %111                       ; Poort_E I/O

                                ;76543210          ; Hulpregel analoog
ADCON0 = %00000001                ; ADCON0 register analoog

                                ;543210           ; Hulpregel analoog
ANSELH = %000000                  ; ANSEL register analoog

;-----
; HIER ONDER DE CODE VOOR HET SERIEEL VERSTUREN VAN DE ANALOGE INGANG.
;-----

Run:
    DATA_UIT = ADIn 0
    DelayMS 20
    RSOut DATA_UIT
GoTo RUN

End

```

Zoals je kan zien is het maar een heel korte code, de code bevat maar zes regels.

```

;-----
; HIER ONDER DE CODE VOOR HET SERIEEL VERSTUREN VAN DE ANALOGE INGANG.
;-----

```

```
RUN:
  DATA_UIT = ADIn 0
  DelayMS 20
  RSOut DATA_UIT
GoTo RUN
```

End

Met dit stukje code kan je dus een potmeter signaal versturen naar Profilab.
We gaan de rest van het programma eens bekijken.

Zoals je kan zien starten we met DEVICE 16F887, dat is dus de controller die we gebruiken.
Daaronder staat de kristal frequentie XTAL 10, daar geven we de waarde van het gebruikte kristal op.
In dit geval is dat 10MHz. Daar onder staat de CONFIG, daar geven we de instellingen op voor de
controller. Deze waardes blijven voor dit type controller het zelfde, die worden niet veranderd.

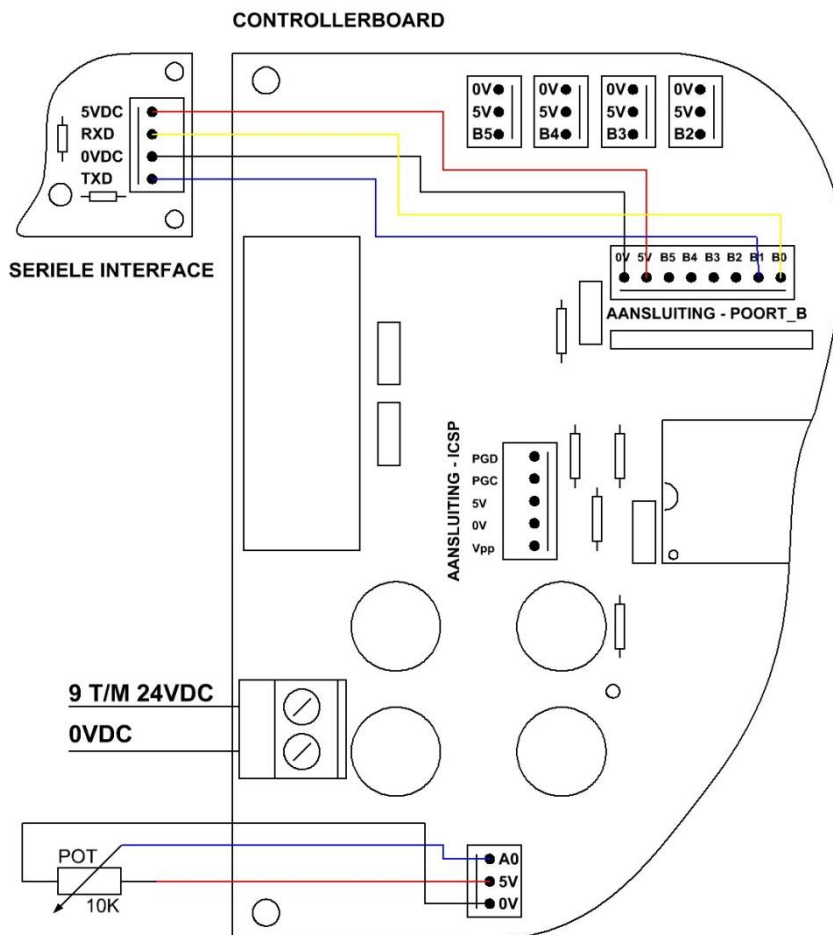
Je kan ook zien dat we de CONFIG beginnen met het commando ASM en eindigen met ENDASM, dit
is een stukje assembler code. Onder de CONFIG staan de DECLARE commando's, hier zeg je tegen de
controller wat de functies moeten worden die de controller uit moet gaan voeren. Dit wordt in het
register van de controller gezet. Hier zeg je dus dat je de analoge poort wil gebruiken, de HPWM
poort, de baudrate voor de seriële overdracht, en welke pinnen er voor de seriële poort worden
gebruikt. Onder het commando DIM worden de variabele gezet, in dit geval noemen we die
DATA_UIT. Dit noemen we zo omdat we data uit gaan sturen maar je kan het ook PIET of JANTJE
noemen natuurlijk, maar deze tekst past beter bij de functie.

Daaronder zetten we het commando CLEAR, dit wist alle ram geheugen van de controller zodat er
geen rare waardes in blijven staan.

Onder CLEAR zetten we de poort instellingen, eerst zetten we alle poorten op nul, dat doen we met
PORTA = %000000, PORTB = %000000, PORTC = %00000000, PORTD = %00000000, en PORTE = %000
En daarna gaan we met TRISA = %111111, TRISB = %000001, TRISC = %00000000,
TRISD = %00000000, en TRISE = %111 de I/O functie instellen. Als we achter TRIS een 0 invullen dan
wordt die pin een uitgang, en als we achter TRIS een 1 invullen dan wordt de pin een ingang. Boven
de poort instelling zie je een hulpregel staat, daarmee kan je zien om welke poort pin het gaat. Als je
bijv. bij PORTA kijkt dan kan je zien dat de meest rechtse pin PORTA.0 is. Daar staat ook een nul in
het blauw boven. En de meest linkse pin is PORTA.5. Op deze manier stel je dus alle poorten in als in
of uitgang.

Daar onder staat de instelling voor de analoge ingangen, dat doe je met ADCON0 en ANSELH. Op
deze manier wordt het register ingesteld voor de analoge poorten. Zoals je aan het stukje code kan
zien gebruiken we de analoge ingang ADIN 0, dit stukje code zegt niet meer dan zet de waarde van
de potmeter in DIM DATA_UIT AS BYTE, wacht 20mS en stuur het via RSOUT DATA_UIT naar de
serieel ingestelde poort PORTB.1. Dat is dus de hele code die nodig is voor dit voorbeeld.

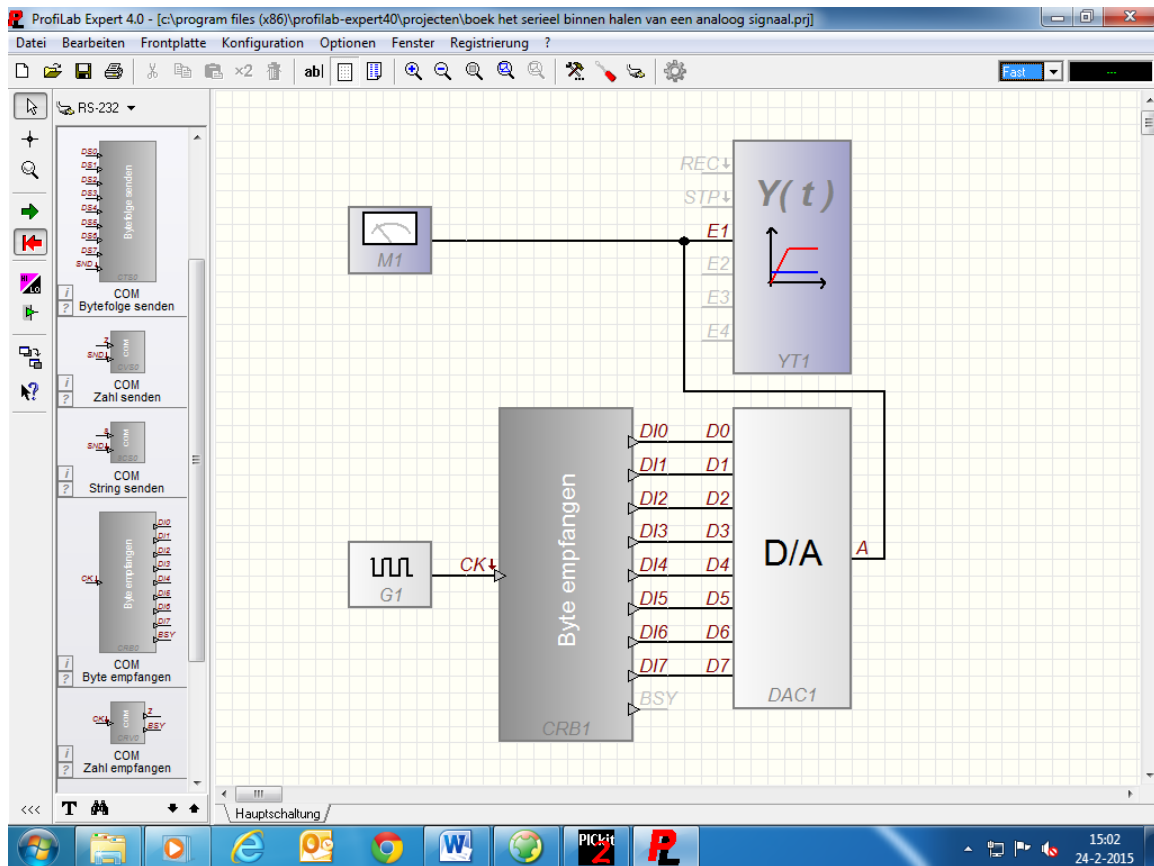
Hieronder een aansluitschema voor de potmeter en de seriële interface.



Op deze manier wordt de seriële interface en de potmeter aangesloten op het controllerboard. Op het schema is te zien dat de seriële interface op poort B zit, PORTB.0 is RXD, en PORTB.1 is TXD. De voedingsspanning voor de seriële interface wordt ook aangesloten, deze is nodig om de optocouplers van spanning te voorzien. Dit is zowel voor TXD als RXD.

De potmeter is op PORTA.0 aangesloten, die is geconfigureerd als analoge ingang op het controllerboard.

We gaan nu een programma maken in ProfiLab om de data binnen te halen, en de waarde van de potmeter te tonen op het beeldscherm van de pc.



We slepen één puls geveer, één analoge meter, één acht bits seriële module, één digitaal/analoog module, en één schrijver module naar het werkblad. De puls geveer G1 komt aan de CK van de CRB1, de waarde van de pulsgeveer zetten we op vijftig (50Hz dus). De DI0 tot en met DI7 van de CRB1 komen één op één aan de digitaal/analoog module DAC1, de waarde van DAC1 stellen we in tussen de 0 en 255 (dit kan weer onder eigenschappen van de module). De analoge uitgang van DAC1 komt aan de analoge meter M1, en aan de E1 ingang van de schrijver YT1. De analoge meter M1 zetten we ook van 0 tot en met 255, en dat doen we ook bij de schrijver YT1. Bij de schrijver en de meter is ook de tekst waarde ingegeven. De log snelheid van de schrijver staat ingesteld op 100mS.

De instelling voor de seriële module CRB1 is: COM 5, 9600 BAUD, DATA 8, PARITY NONE, STOP 1. Dit kan weer onder eigenschappen van de module CRB1 ingevuld worden. De compoort staat hier op vijf, maar dat ligt er helemaal aan op welke usb poort het zit, dat kan anders zijn natuurlijk. Dus dat moet men zelf ter plaatse even bekijken.

De indeling van de frontplaat heb ik zo gemaakt.

Zie Blz. 64.



Op de front plaat is goed te zien wat de schrijver geschreven heeft, waar de dikkere lijnen staan, (bij 1 : 00) heb ik de potmeter snel heen en weer gedraaid. Dus daar schrijft de schrijver wat vetter, de waarde wordt ook weergegeven op de analoge meter boven de schrijver.

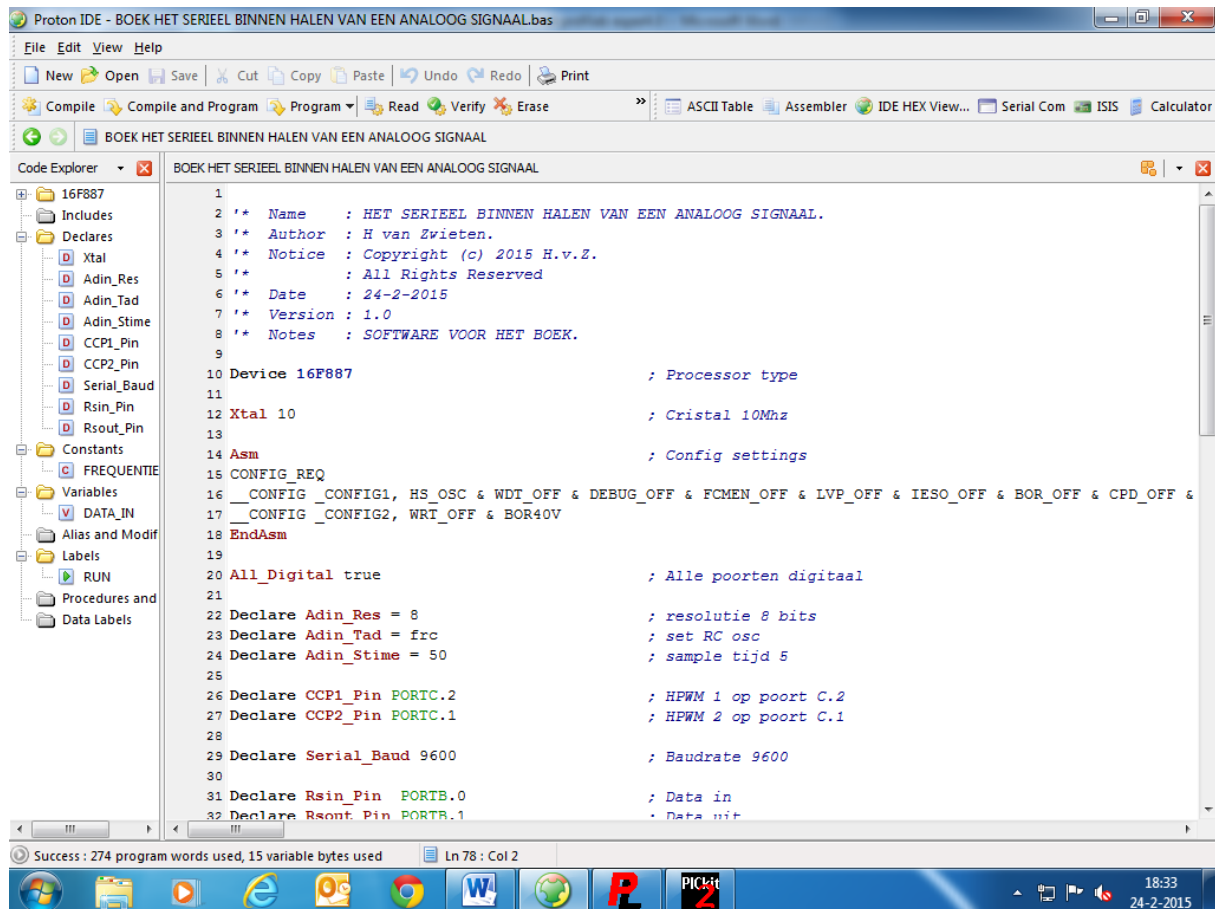
Met het rode knopje start je de schrijver, en met het vierkante knopje stop je de schrijver. Dan heb je nog een knopje voor data printen, data openen, data opslaan, en een instelling voor de snelheid voor opnemen van de data. De knopjes onder zijn voor het inzoomen horizontaal, het uitzoomen horizontaal, het inzoomen verticaal, het uitzoomen verticaal, het zoomen voor het hele plaatje, de zoom reset, het snel doorspoelen naar links, het snel doorspoelen naar rechts, en de knop om naar eigenschappen te gaan van de schrijver.

Als je de geregistreeerde data op wil slaan, dan kan dat alleen als het programma loopt. En dat geldt ook voor als je opgeslagen data wil laden.

Dit is één van de manieren om data zichtbaar te maken, maar dat kan ook anders natuurlijk. Dus daar kan zelf weer mee geëxperimenteerd worden.

6.2: Het serieel uitsturen van een analogo signaal

We gaan nu het tegen over gestelde doen, we gaan vanuit Profilab een analogo signaal versturen naar de controller. We beginnen weer met het basic programma.



```
1
2  /* Name      : HET SERIEEL BINNEN HALEN VAN EEN ANALOOG SIGNAAL.
3  /* Author    : H van Zwieten.
4  /* Notice    : Copyright (c) 2015 H.v.Z.
5  /*          : All Rights Reserved
6  /* Date      : 24-2-2015
7  /* Version   : 1.0
8  /* Notes    : SOFTWARE VOOR HET BOEK.
9
10 Device 16F887                ; Processor type
11
12 Xtal 10                       ; Cristal 10Mhz
13
14 Asm                           ; Config settings
15 CONFIG_REQ
16 __CONFIG _CONFIG1, HS_OSC & WDT_OFF & DEBUG_OFF & FCMEN_OFF & LVP_OFF & IESO_OFF & BOR_OFF & CPD_OFF &
17 __CONFIG _CONFIG2, WRT_OFF & BOR40V
18 EndAsm
19
20 All_Digital true             ; Alle poorten digitaal
21
22 Declare Adin_Res = 8         ; resolutie 8 bits
23 Declare Adin_Tad = frc       ; set RC osc
24 Declare Adin_Stime = 50     ; sample tijd 5
25
26 Declare CCP1_Pin PORTC.2    ; HPWM 1 op poort C.2
27 Declare CCP2_Pin PORTC.1    ; HPWM 2 op poort C.1
28
29 Declare Serial_Baud 9600    ; Baudrate 9600
30
31 Declare Rsin_Pin PORTB.0     ; Data in
32 Declare Rsoout_Pin PORTR.1   ; Data uit
```

We gaan het programma hier onder weer helemaal plaatsen.

```
/* Name      : HET SERIEEL BINNEN HALEN VAN EEN ANALOOG SIGNAAL.
/* Author    : H van Zwieten.
/* Notice    : Copyright (c) 2015 H.v.Z.
/*          : All Rights Reserved
/* Date      : 24-2-2015
/* Version   : 1.0
/* Notes    : SOFTWARE VOOR HET BOEK.
```

```
Device 16F887                ; Processor type
```

```
Xtal 10                       ; Cristal 10Mhz
```

```
Asm                           ; Config settings
```

```
CONFIG_REQ
```

```
__CONFIG _CONFIG1, HS_OSC & WDT_OFF & DEBUG_OFF & FCMEN_OFF & LVP_OFF & IESO_OFF & BOR_OFF & CPD_OFF & CP_OFF & MCLR_OFF & PWRTE_ON
```

```
__CONFIG _CONFIG2, WRT_OFF & BOR40V
```

```
EndAsm
```

```

All_Digital true ; Alle poorten digitaal

Declare Adin_Res = 8 ; resolutie 8 bits
Declare Adin_Tad = frc ; set RC osc
Declare Adin_Stime = 50 ; sample tijd 5

Declare CCP1_Pin PORTC.2 ; HPWM 1 op poort C.2
Declare CCP2_Pin PORTC.1 ; HPWM 2 op poort C.1

Declare Serial_Baud 9600 ; Baudrate 9600

Declare Rsin_Pin PORTB.0 ; Data in
Declare Rsout_Pin PORTB.1 ; Data uit

Symbol FREQUENTIE = 1000 ; Frequentie is 1KHz

Dim DATA_IN As Byte ; Data serieelepoort compoort

Clear ; Wis geheugen

;543210 ; Hulpregel poort A
PORTA = %000000 ; Maak poort A laag
TRISA = %111111 ; Poort_A I/O

;543210 ; Hulpregel poort B
PORTB = %000000 ; Maak poort B laag
TRISB = %000001 ; Poort_B I/O

;76543210 ; Hulpregel poort C
PORTC = %00000000 ; Maak poort C laag
TRISC = %00000000 ; Poort_C I/O

;76543210 ; Hulpregel poort D
PORTD = %00000000 ; Maak poort D laag
TRISD = %00000000 ; Poort_D I/O

;210 ; Hulpregel poort E
PORTE = %000 ; Maak poort E laag
TRISE = %111 ; Poort_E I/O

;76543210 ; Hulpregel analoog
ADCON0 = %00000001 ; ADCON0 register analoog

;543210 ; Hulpregel analoog
ANSELH = %000000 ; ANSEL register analoog

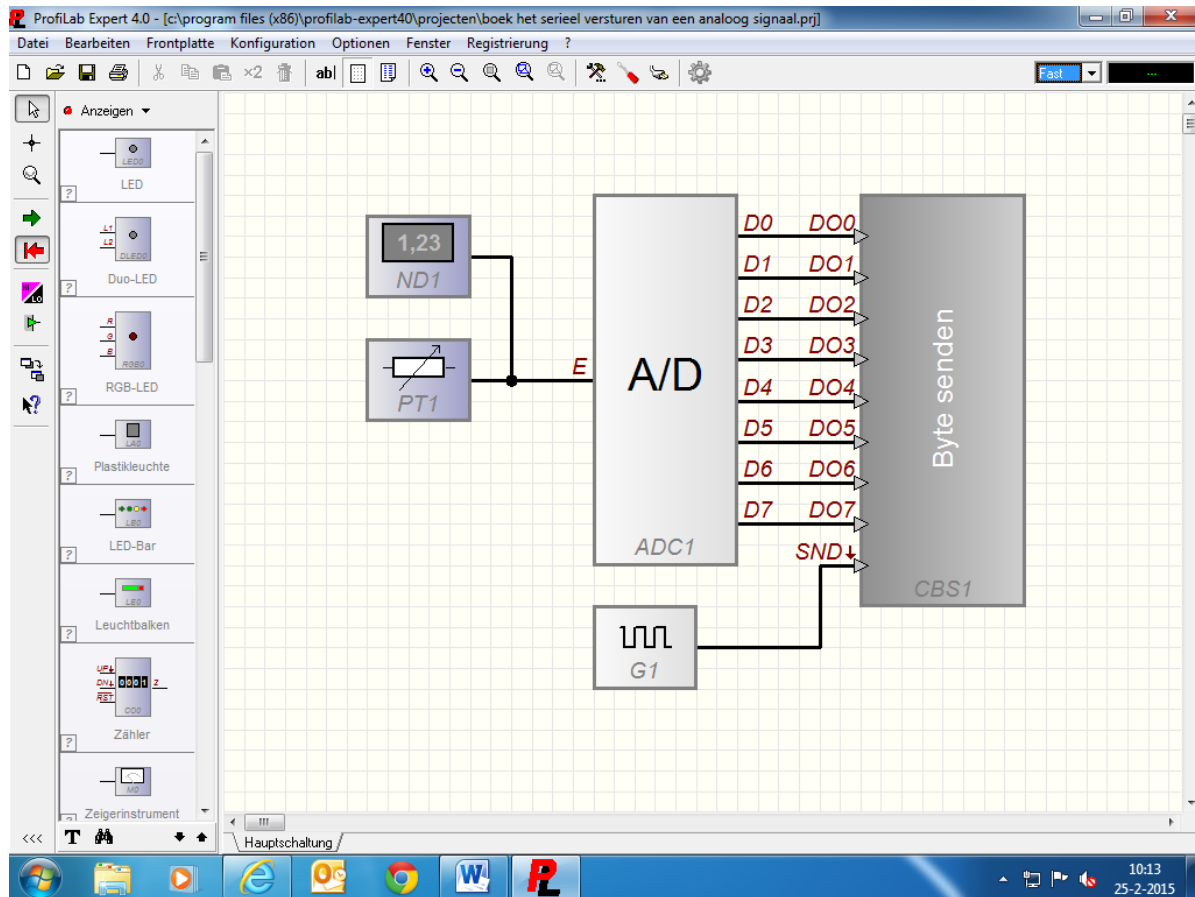
;-----
; HIER ONDER DE CODE VOOR HET SERIEEL BINNEN HALEN VAN EEN ANALOOG SIGNAAL.
;-----

RUN:
    DATA_IN = RSIn
    DelayMS 20
    HPWM 2, DATA_IN, FREQUENTIE
GoTo RUN

End

```

Zoals je kan zien is er niet veel veranderd in het programma, SYMBOL FREQUENTIE = 1000, DIM DATA_IN AS BYTE, HPWM, en RSIN is toegevoegd. De code is weer net zo kort als bij het eerste voorbeeld. De data uit RSIN wordt in de DATA_IN variabele gezet, en die data wordt weer gebruikt bij de HPWM functie. De waarde bij SYMBOL FREQUENTIE = 1000 genereert een waarde van 1KHz, dat is dus de frequentie voor het pwm signaal. Tot zover het basic programma dat is nu klaar. We gaan nu het Profilab programma maken waarmee de data verstuurd kan worden.



We slepen één potmeter, één puls geveer, één display, één analoog / digitaal module, en één acht bit seriële module naar het werkblad. De potmeter komt aan de E ingang van de analoog/digitaal module ADC1 en aan het display ND1, de uitgangen van de analoog/digitaal module ADC1 komen één op één op de ingangen van de seriële module CBS1. Dus D0 aan DO0 tot en met D7 aan DO7. De waarde van de potmeter PT1, analoog/digitaal module ADC1, en het display ND1 stellen we weer in op 0 tot 255.

De waarde van de puls geveer stellen we in op tien (10Hz dus). De acht bits seriële module CBS1 stellen we weer in op COM5, 9600 BAUD, DATA 8, PARITY NONE, STOP 1. Maar COM5 kan natuurlijk weer anders zijn, dat ligt weer aan de gebruikte usb of com poort.

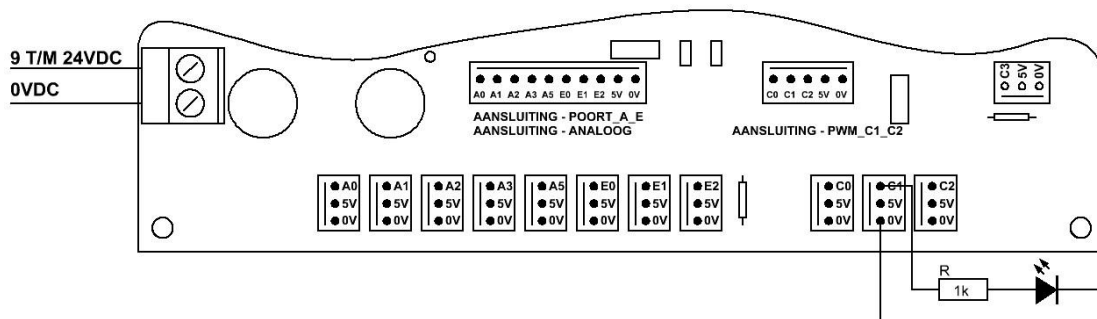
De indeling van de frontplaat heb ik zo gemaakt.

Zie Blz. 68.

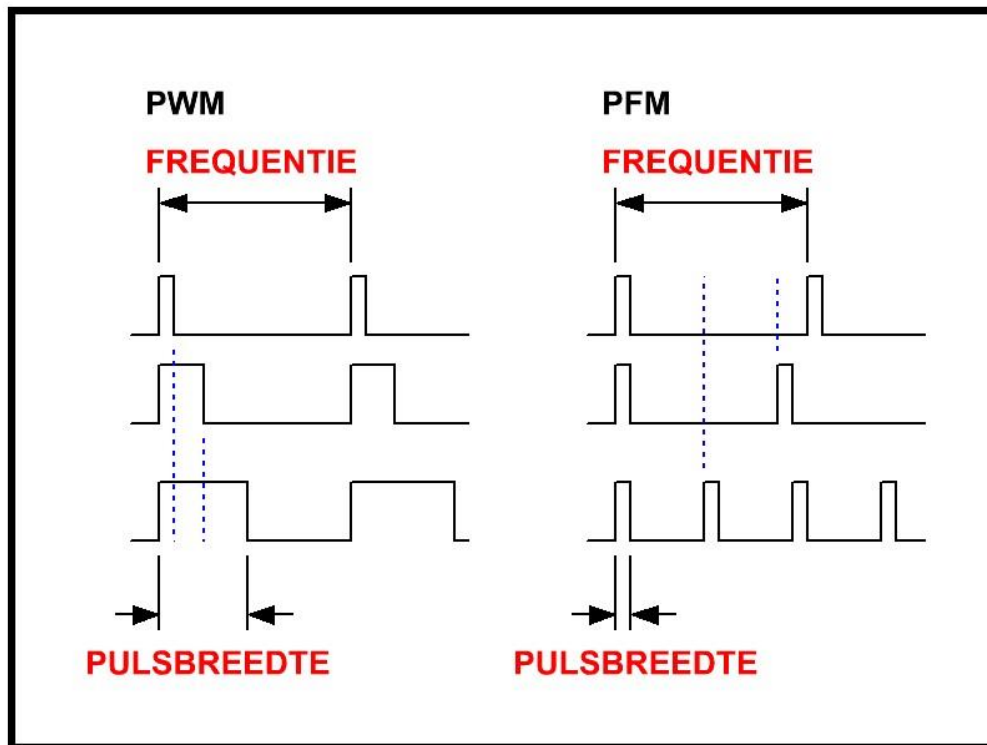


Het enige bediening element is hier de potmeter. Als het Profilab programma is opgestart en de seriële poort is aangesloten op het controllerboard, dan kan je via de potmeter de HPWM poort PORTC.1 bedienen. De HPWM poort zal dan een pwm signaal uitsturen met een waarde tussen de 0 en 255. Die waarde wordt ook op het display getoond boven de potmeter. Als men nu een led aansluit via een voorschakelweerstand van 1K, dan zal je zien dat je deze led kan dimmen. De pwm functie gaan we later voor nog meer voorbeelden gebruiken, maar dat komt later aan bod.

Hier onder een schema hoe je de led aan moet sluiten op de pwm poort.



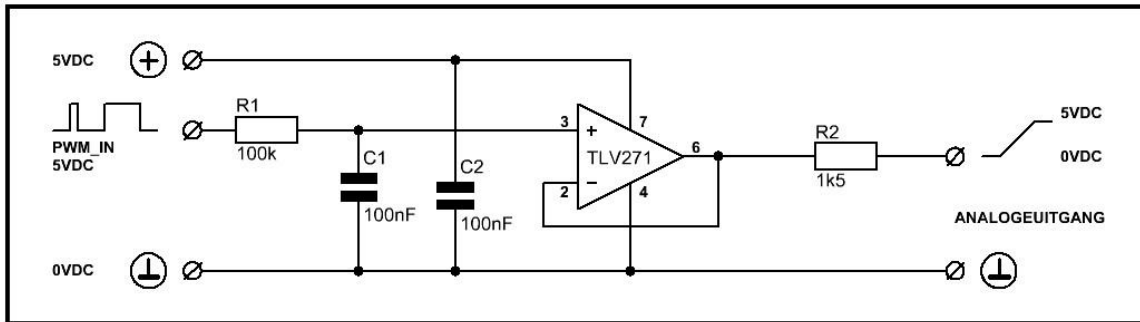
We gaan nog even verder op de pwm functie in. Je hebt twee variaties, dat is pwm en pfm. Bij pwm blijft de frequentie gelijk en verandert de puls breedte, en bij pfm verandert de frequentie en blijft de puls breedte gelijk. Hier onder een voorbeeld van hoe zoiets eruit ziet.



Die functie kan je softwarematig aanpassen in het basic programma. Als je naar het commando **HPWM 2, DATA_IN, FREQUENTIE** kijkt, dan zie je dat je achter HPWM twee variabelen of een vaste waarde in kan geven. Als je op de eerste plek een variabele waarde in geeft en op de tweede plek een vaste waarde, dan heb je een pwm signaal. Als je op de eerste plek een vaste waarde ingeeft en op de tweede plek een variabele waarde ingeeft, dan heb je een pfm signaal. Deze waarden kunnen variëren tussen de 0 en 255, acht bits dus.

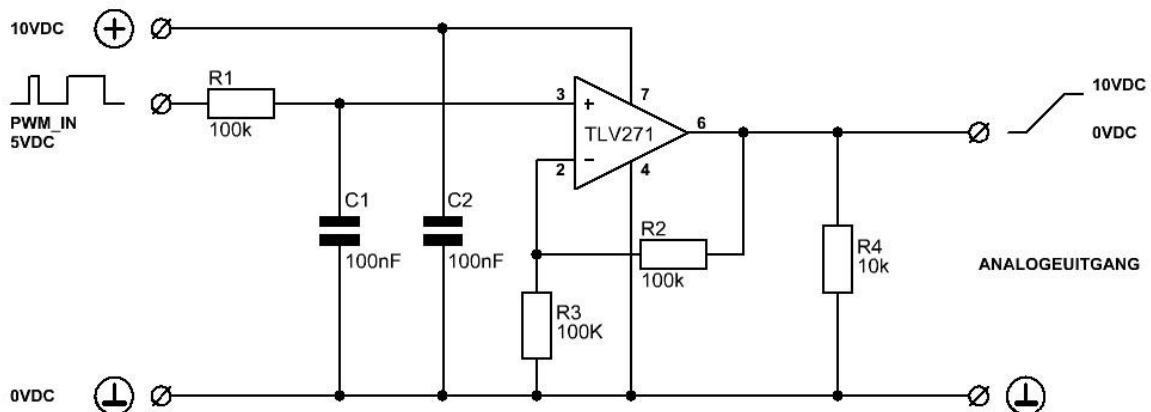
Met dat zelfde pwm signaal kunnen we ook een analoog uitgang signaal maken, dan kan je doen door er een opamp achter te plaatsen.

Zie voorbeeld schema op Blz. 70.



Op deze manier kan je een 5V pwm signaal omzetten naar een uitgangspanning van 0 tot 5VDC. Je kan ook het 5V pwm signaal omzetten naar een hogere spanning bijv. naar 10 of 12VDC, maar dan moet de opamp ook met die spanning gevoed worden. En er komt dan een weerstand netwerk bij om het signaal te versterken. Dit kan bijv. makkelijk zijn om proportionele kleppen aan te sturen via pwm, of andere zaken die een analoge spanning nodig hebben van bijv. 0 tot 10 of 0 tot 12VDC.

Hier onder nog een voorbeeld schema on van een 5V pwm signaal een 0 tot 10V analogo signaal te maken.



6.3: Het serieel aansturen van uitgangen

We gaan nu met Profilab de uitgangen van de controller aansturen, we beginnen weer met het basic programma. Ik ga nu niet meer het werkblad van Proton laten zien, want dan is maar een heel klein stukje van de programma code te zien, ik plaats dus vanaf nu alleen de basic code.

```

'* Name      : BOEK HET SERIEEL AANSTUREN VAN UITGANGEN.
'* Author    : H van Zwieten.
'* Notice    : Copyright (c) 2015 H.v.Z.
'*           : All Rights Reserved
'* Date      : 25-2-2015
'* Version   : 1.0
'* Notes     : SOFTWARE VOOR HET BOEK.

Device 16F887                               ; Processor type

Xtal 10                                     ; Cristal 10Mhz

Asm                                         ; Config settings
CONFIG_REQ
__CONFIG __CONFIG1, HS_OSC & WDT_OFF & DEBUG_OFF & FCMEN_OFF & LVP_OFF &
IESO_OFF & BOR_OFF & CPD_OFF & CP_OFF & MCLRRE_OFF & PWRTE_ON
__CONFIG __CONFIG2, WRT_OFF & BOR40V
EndAsm

All_Digital true                           ; Alle poorten digitaal

Declare Adin_Res = 8                        ; Resolutie 8 bits
Declare Adin_Tad = frc                      ; set RC osc
Declare Adin_Stime = 50                    ; sample tijd 5

Declare CCP1_Pin PORTC.2                   ; HPWM 1 op poort C.2
Declare CCP2_Pin PORTC.1                   ; HPWM 2 op poort C.1

Declare Serial_Baud 9600                   ; Baudrate 9600

Declare Rsin_Pin PORTB.0                   ; Data in
Declare Rsout_Pin PORTB.1                  ; Data uit

Symbol UIT_1 = PORTA.0                     ; Uitgang één
Symbol UIT_2 = PORTA.1                     ; Uitgang twee
Symbol UIT_3 = PORTA.2                     ; Uitgang drie
Symbol UIT_4 = PORTA.3                     ; Uitgang vier
Symbol UIT_5 = PORTA.5                     ; Uitgang vijf
Symbol UIT_6 = PORTE.0                     ; Uitgang zes
Symbol UIT_7 = PORTE.1                     ; Uitgang zeven
Symbol UIT_8 = PORTE.2                     ; Uitgang acht

Dim DATA_IN As Byte                       ; Data serieelepoort compoort

Clear                                       ; Wis geheugen

;543210                                    ; Hulpregel poort A
PORTA = %000000                            ; Maak poort A laag
TRISA = %000000                            ; Poort_A I/O

;543210                                    ; Hulpregel poort B
PORTB = %000000                            ; Maak poort B laag
TRISB = %000001                            ; Poort_B I/O

;76543210                                  ; Hulpregel poort C
PORTC = %00000000                          ; Maak poort C laag
TRISC = %00000000                          ; Poort_C I/O

```

```

        ;76543210                ; Hulpregel poort D
PORTD  = %00000000              ; Maak poort D laag
TRISD  = %00000000              ; Poort_D I/O

        ;210                    ; Hulpregel poort E
PORTE  = %000                  ; Maak poort E laag
TRISE  = %000                  ; Poort_E I/O

        ;76543210                ; Hulpregel analoog
ADCON0 = %00000001            ; ADCON0 register analoog

        ;543210                  ; Hulpregel analoog
ANSELH = %000000              ; ANSEL register analoog

```

```

;-----
; HIER ONDER DE CODE VOOR HET SERIEEL BINNEN HALEN VAN DE DATA.
;-----

```

```
DATA_IN = 0
```

```

UIT_1 = 0
UIT_2 = 0
UIT_3 = 0
UIT_4 = 0
UIT_5 = 0
UIT_6 = 0
UIT_7 = 0
UIT_8 = 0

```

```
RUN:
```

```

DATA_IN = RSIn
DelayMS 20

```

```

If DATA_IN = 1 Then
  Toggle UIT_1
  DelayMS 50
EndIf

```

```

If DATA_IN = 2 Then
  Toggle UIT_2
  DelayMS 50
EndIf

```

```

If DATA_IN = 3 Then
  Toggle UIT_3
  DelayMS 50
EndIf

```

```

If DATA_IN = 4 Then
  Toggle UIT_4
  DelayMS 50
EndIf

```

```

If DATA_IN = 5 Then
  Toggle UIT_5
  DelayMS 50
EndIf

```



```

If DATA_IN = 6 Then
  Toggle UIT_6
  DelayMS 50
EndIf

If DATA_IN = 7 Then
  Toggle UIT_7
  DelayMS 50
EndIf

If DATA_IN = 8 Then
  Toggle UIT_8
  DelayMS 50
EndIf
GoTo RUN

End

```

De code wordt nu een stukje langer zoals je kan zien wat er nu bijgekomen is onder SYMBOL is, UIT_1 tot en met UIT_8. En daarachter staan ook de poort namen die gebruikt zijn, PORTA.0, PORTA.1, PORTA.2, PORTA.3, PORTA.5, PORTE.0, PORTE.1, en PORTE.2. Deze poorten zitten allemaal naast elkaar op het controllerboard. TRISA en TRISE is ook aangepast die staan nu op %000000 en op %000, dit is gedaan omdat we die poorten nu als uitgang gaan gebruiken. Ook kan je zien dat het commando Toggle nu wordt gebruikt dat commando doet niet anders dan, als er een puls binnen komt van Profilab wordt de desbetreffende uitgang op het controllerboard hoog of laag gemaakt.

Voorbeeld: Onder Profilab wordt er onder bijv. knop één de waarde één verstuurd. Die waarde komt in de variabele DATA_IN te staan en wordt vergeleken in IF DATA_IN = 1 THEN TOGGLE UIT_1, op dat moment wordt uitgang één hoog gemaakt. Wordt er nogmaals op knop één gedrukt, dan wordt de uitgang weer laag. Dit gebeurt bij alle acht uitgangen op het controllerboard.

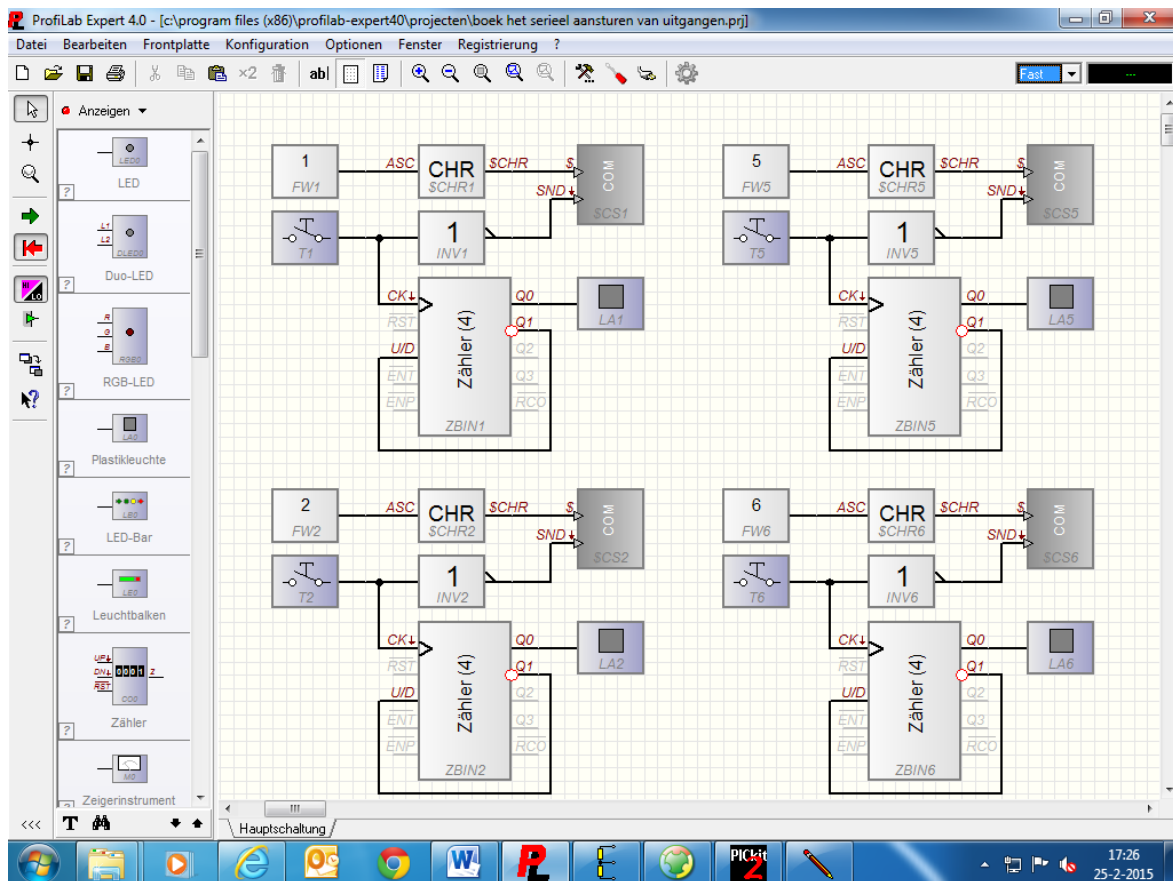
De DELAYMS 50 zorgt ervoor dat de uitgang niet gaat staan klapperen, (ongewild hoog en laag wordt) De ENDIF zorgt ervoor dat de IF – THEN functie afgesloten wordt.

Net voordat het programma begint zie je dat de DATA_IN en UIT_1 tot en met UIT_8 op nul gezet wordt. Dit is gedaan om er zeker van te zijn dat de data en de uitgangen op nul staan voordat het programma de lus in gaat, anders zou het kunnen gebeuren dat er al een uitgang hoog staat zonder dat er een knop ingedrukt is. En dat wil je ten alle tijden voorkomen.

We gaan nu het Profilab programma maken. Ik laat maar een aantal gemaakte modules zien, omdat het anders niet goed duidelijk is hoe de componenten aan elkaar gekoppeld zijn.

Ik laat de modules zien waar uitgang één, twee, vijf, en zes op staan. De modules voor uitgang drie, vier, zeven en acht zijn precies het zelfde. Als je er één maakt kan je ze eigenlijk gewoon kopiëren, je hoeft dan alleen de waardes maar aan te passen.

Zie voorbeeld op Blz. 74.



We slepen één vaste waarde module, één drukknop, één ascii naar string module, één inverter, één teller module, één controle lamp, en één seriële module naar het werkblad. We verbinden de vaste waarde module FW1 met de ascii naar string module \$CHR1, en de string uitgang van de \$CHR1 verbinden we met de string ingang van de seriële module \$CS1. De drukknop T1 komt aan de inverter INV1 en aan de CK ingang van de teller module ZBIN1. De uitgang van de inverter module INV1 komt aan SND ingang van de seriële module \$CS1. Op de teller module ZBIN1 wordt de Q0 uitgang verbonden met de lamp LA1, en de Q1 uitgang wordt verbonden met de U/D ingang van de teller module ZBIN1. Ook wordt de Q1 pin van de teller ZBIN1 geïnverteerd, dat kan gedaan worden door op de pin te klikken. Als det goed gedaan is verschijnt er een rondje bij de desbetreffende pin. Bij de vaste waarde module vullen we één in, als er nu op knop T1 gedrukt wordt zal het getal één serieel verzonden worden door de seriële module \$CS1. De waarde die in de seriële module komt te staan is weer COM5, 9600 BAUD, DATA 8, PARITY NONE, en STOP 1. Nu hebben we het gedeelte voor één drukknop af. Als je van dat gedeelte nu een kopie maakt, dan kan je met de functie invoegen acht van deze gedeeltes maken. Dan hoef je alleen de waardes bij de vaste waarde module FW2 tot en met FW8 aan te pasen, voor de rest wordt alles mee genomen. Ook de waardes die je in de seriële module \$CS1 gezet hebt, daar hoef je niets meer in te geven. Als je dat gedaan hebt en je gaat daarna naar de frontplaat, dan zie je maar één drukknop en één lamp staan. Dat lijkt zo, maar er staan er van elk acht. Ze staan allemaal op elkaar, als je er één versleept zal je zien dat er een ander onder vandaan komt.

De indeling van de frontplaat heb ik zo gemaakt.

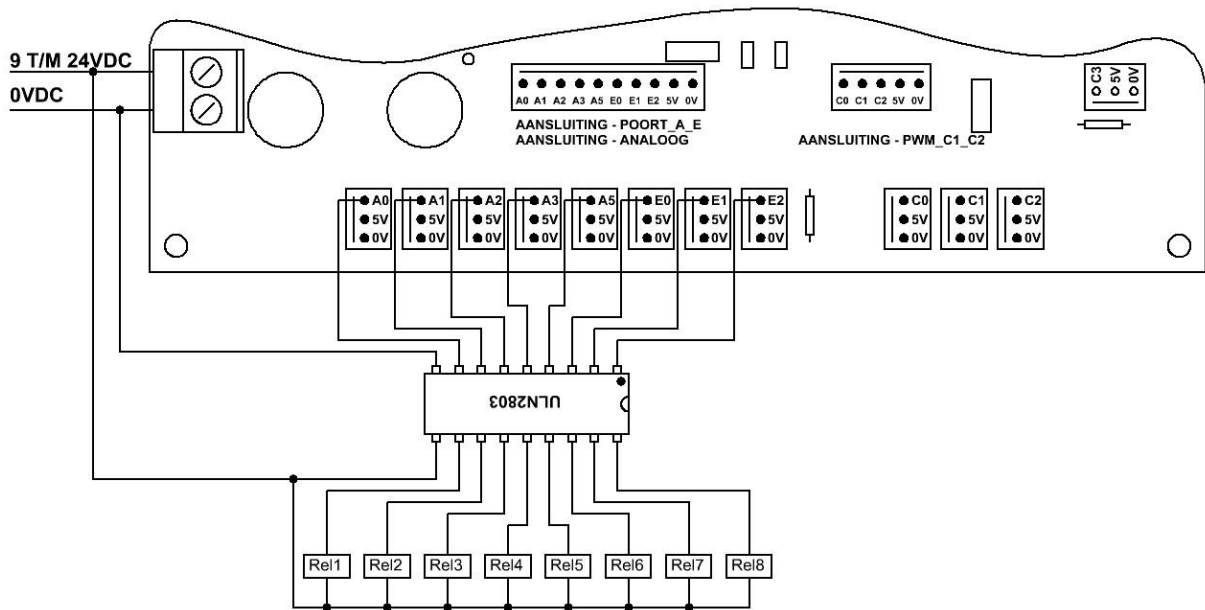


Zoals je kan zien hebben we acht drukknoppen en acht lampen. Als je nu op bijv. knop `UITGANG_1` drukt gaat de lamp boven de drukknop aan, en wordt het getal één verzonden over de usb poort COM5 en de software in de controller zorgt ervoor dat PORTA.0 hoog wordt zoals we besproken hebben in het basic programma. Wordt er nogmaals op de knop gedrukt dan wordt PORTA.0 weer laag.

De lamp zit op de teller module ZBIN1 aan de Q0 uitgang. Als je op knop `UITGANG_1` drukt krijgt de teller een puls op de klok ingang en zorgt er voor dat Q0 hoog wordt, (de lamp gaat dus aan) als er nogmaals gedrukt wordt gaat de lamp weer uit.

Op de uitgangen van het controllerboard kan je een paar leds plaatsen om de werking te simuleren. Dit kan op de zelfde manier zoals in het voorbeeld met het pwm signaal, met een voorschakel weerstand van 1K in serie met de led dus. Je kan bijv. ook een ULN2803 driver op het controllerboard aan sluiten, dan kan je daar bijv. weer relais mee schakelen.

Zie schema op Blz. 76.



Op deze manier kan je de relais aansluiten op het controllerboard. Het is nu op de onderste connectoren aangesloten, maar het kan ook op de tien polige connector die er boven staat. Dit is bijv. handig als je een relais kaart gebruikt, dan heb je maar één connector nodig. Het puntje dat bij de ULN2803 staat is pin één, pin negen is de gnd en pin tien is de plus van het ic. De spoelspanning voor de relais komt van de voedingsspanning van het controllerboard. Je moet er wel op letten dat als je bijv. 12VDC relais gebruikt, dat de voedingsspanning van het controllerboard ook 12VDC is. Normaal staan er ook vrijloop diodes over de relais spoelen, dat is nu niet nodig omdat de diodes al in de ULN2803 zitten. De ULN2803 mag max zo'n 100mA per pin schakelen, dat is voor de meeste relais en bijv. spoelen van kleppen meer dan voldoende. Meer informatie is in de datasheet van de ULN2803 te vinden. Tot zover dit voorbeeld.

6.4: Het serieel binnen halen van ingangen

We gaan nu de ingangen van de controller binnen halen, en verwerken in Profilab. We gaan ook een paar andere type controllers erbij halen in dit voorbeeld, maar later daarover meer.

We beginnen weer met het basic programma voor de controller.

```
'* Name      : BOEK HET SERIEEL BINNEN HALEN VAN INGANGEN.
'* Author    : H van Zwieten.
'* Notice    : Copyright (c) 2015 H.v.Z.
'*           : All Rights Reserved
'* Date      : 26-2-2015
'* Version   : 1.0
'* Notes     : SOFTWARE VOOR HET BOEK.
```

```
Device 16F887 ; Processor type
```

```
Xtal 10 ; Cristal 10Mhz
```

```

Asm                                     ; Config settings
CONFIG_REQ
__CONFIG __CONFIG1, HS_OSC & WDT_OFF & DEBUG_OFF & FCMEN_OFF & LVP_OFF &
IESO_OFF & BOR_OFF & CPD_OFF & CP_OFF & MCLR_OFF & PWRTE_ON
__CONFIG __CONFIG2, WRT_OFF & BOR40V
EndAsm

All_Digital true                         ; Alle poorten digitaal

Declare Adin_Res = 8                       ; Resolutie 8 bits
Declare Adin_Tad = frc                     ; set RC osc
Declare Adin_Stime = 50                   ; sample tijd 5

Declare CCP1_Pin PORTC.2                 ; HPWM 1 op poort C.2
Declare CCP2_Pin PORTC.1                 ; HPWM 2 op poort C.1

Declare Serial_Baud 9600                 ; Baudrate 9600

Declare Rsin_Pin PORTB.0                 ; Data in
Declare Rsout_Pin PORTB.1                ; Data uit

Symbol IN_1 = PORTA.0                     ; Ingang één
Symbol IN_2 = PORTA.1                     ; Ingang twee
Symbol IN_3 = PORTA.2                     ; Ingang drie
Symbol IN_4 = PORTA.3                     ; Ingang vier

Clear                                     ; Wis geheugen

;543210                                     ; Hulpregel poort A
PORTA = %000000                             ; Maak poort A laag
TRISA = %111111                             ; Poort_A I/O

;543210                                     ; Hulpregel poort B
PORTB = %000000                             ; Maak poort B laag
TRISB = %000001                             ; Poort_B I/O

;76543210                                   ; Hulpregel poort C
PORTC = %00000000                          ; Maak poort C laag
TRISC = %00000000                          ; Poort_C I/O

;76543210                                   ; Hulpregel poort D
PORTD = %00000000                          ; Maak poort D laag
TRISD = %00000000                          ; Poort_D I/O

;210                                        ; Hulpregel poort E
PORTE = %000                                ; Maak poort E laag
TRISE = %000                                ; Poort_E I/O

;76543210                                   ; Hulpregel analoog
ADCON0 = %00000001                         ; ADCON0 register analoog

;543210                                     ; Hulpregel analoog
ANSELH = %000000                           ; ANSEL register analoog

;-----
; HIER ONDER DE CODE VOOR HET SERIEEL VERSTUREN VAN DE DATA.
;-----

```

```

RUN:
  If IN_1 = 1 Then
    RSOut 1
    DelayMS 100
    If IN_1 = 0 Then
      RSOut 0
      DelayMS 100
    EndIf
  EndIf

  If IN_2 = 1 Then
    RSOut 2
    DelayMS 100
    If IN_2 = 0 Then
      RSOut 0
      DelayMS 100
    EndIf
  EndIf

  If IN_3 = 1 Then
    RSOut 3
    DelayMS 100
    If IN_3 = 0 Then
      RSOut 0
      DelayMS 100
    EndIf
  EndIf

  If IN_4 = 1 Then
    RSOut 4
    DelayMS 100
    If IN_4 = 0 Then
      RSOut 0
      DelayMS 100
    EndIf
  EndIf
GoTo RUN

End

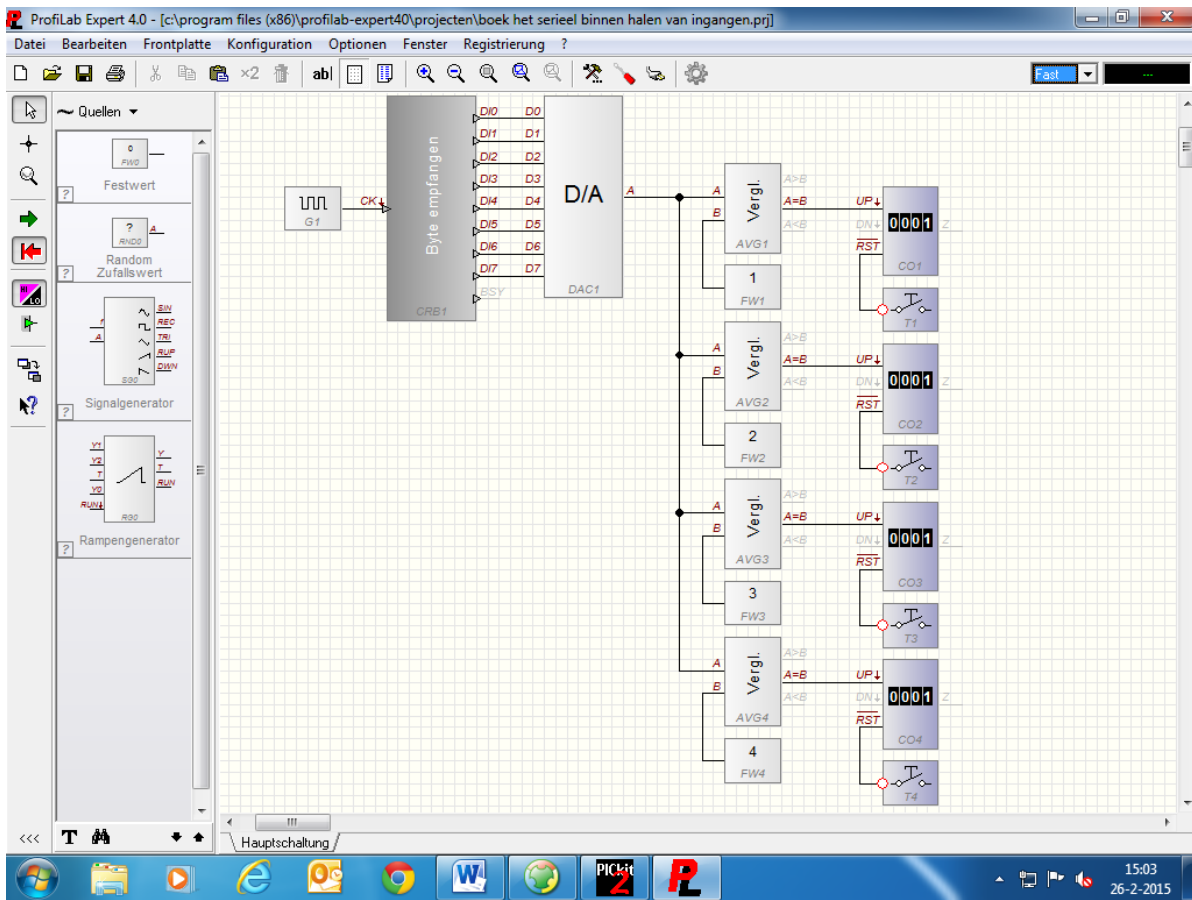
```

De code is weer wat aangepast zoals je kan zien. PORTA staat nu als ingang, (zoals je kan zien bij TRISA = %11111) omdat we daar de signalen van uit gaan lezen in dit voorbeeld. Even wat uitleg over de code. Als ingang IN_1 = 1 THEN RSOUT 1 en als IN_1 = 0 THEN RSOUT 0. Dit wil dus niet anders zeggen dat als ingang één hoog gemaakt wordt dat er een één serieel verzonden wordt door de usb poort COM5, en als ingang één laag is wordt er serieel een nul verzonden. Dit is dus alles wat de code doet, dat geldt voor IN_1, IN_2, IN_3, en IN_4.

Deze code wordt weer binnen gehaald door Profilab, en daar weer verder verwerkt in het programma.

We gaan nu het Profilab programma maken. Ik heb een voorbeeld gemaakt met tellers, dus zodra er een ingang op het controllerboard hoog gemaakt wordt zal de teller optellen. Zoals je in het basic programma kon zien gebruiken we vier ingangen van de controller.

Hieronder het Profilab programma.



We slepen één puls geveer, één acht bit seriële module, één digitaal/anaaloo module, vier vergelijker modules, vier vaste waarde modules, vier drukknoppen, en vier teller modules naar het werkblad. De puls geveer G1 komt aan CK ingang van de seriële module CRB1. De D10 tot en met D17 van de CRB1 komen één op één aan de digitaal/anaaloo module DAC1. De uitgang A van de digitaal/anaaloo module komt aan alle A ingangen van de vergelijker modules AVG1, AVG2, AVG3, en AVG4. De vaste waarde modules FW1, FW2, FW3, en FW4 komen aan de B ingang van de vergelijkers. De A = B uitgangen van de vergelijkers komen aan de UP ingang van de tellers CO1, CO2, CO3, en CO4. De drukknoppen T1, T2, T3, en T4, komen aan de reset ingang van de tellers.

De waarde van de puls geveer G1 zetten we op honderd, op 100Hz dus. De waarde van de digitaal/anaaloo module stellen we in tussen 0 en 255. De waarde van de vaste waarde modules is één tot en met vier. De waarde die in de seriële module komt te staan is weer COM5, 9600 BAUD, DATA 8, PARITY NONE, en STOP 1.

Je kan weer één setje maken van één vergelijker module, één vaste waarde module, en één teller module. Als je die met elkaar verbonden hebt kan je die weer kopiëren en invoegen.

We gaan de frontplaat weer samen stellen.

Zie voorbeeld op Blz. 80.

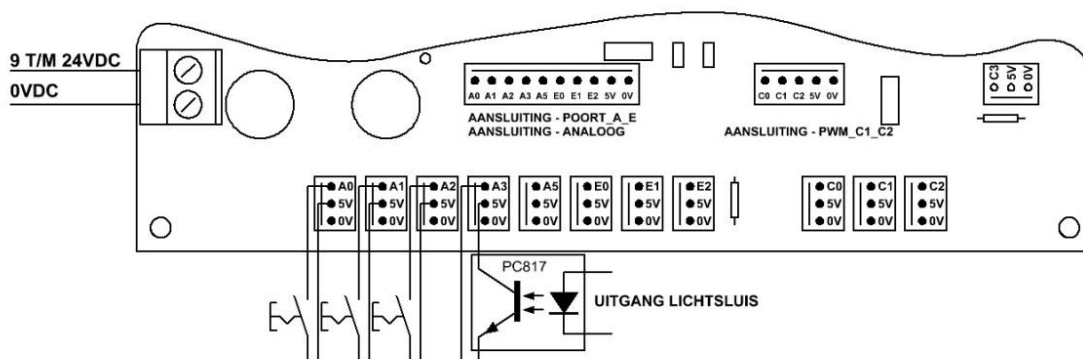
De indeling van de frontplaat heb ik zo gemaakt



de drukknoppen zijn van de tekst RESET voorzien. De tellers bestaan uit vier cijfers.

Als er nu serieel data binnen komt, (dus bijv. een één) dan zal TELLER_1 op één gaan staan. En bij een volgen de puls op twee, enz., enz. Via de reset knoppen onder de tellers, kunnen de tellers op nul gezet worden. Je kan hier weer van alles bij maken als je wilt, zo kan je bijv. de data weer naar een tabel schrijven, of naar Word of Exel. Hier kan weer mee geëxperimenteerd worden.

Hieronder het aansluit schema.



Zo kan je bijv. via drukknoppen de ingangen op het controllerboard bedienen, en de data versturen. Maar zoals je ziet staat er ook een optocoupler in het schema, Dat is de uitgang van een lichtsluis. We gaan een infrarood lichtsluis maken voor één van de teller functies die in het Profilab programma staan.

In het begin van dit voorbeeld had ik het er over dat we nog een paar andere type controllers gingen gebruiken, dat is dus om de lichtsluis mee te maken. Later plaats ik het schema voor de lichtsluis. De controllers die we daar voor gaan gebruiken zijn de 12F683 voor het zend gedeelte, en de 12F675 voor het ontvang gedeelte van de lichtsluis.

Eerst gaan we het basic programma maken voor het zend gedeelte van de lichtsluis.

```

'* Name      : BOEK IR ZENDER VOOR LICHTSLUIS.

'* Author    : H van Zwieten.
'* Notice    : Copyright (c) 2015 H.v.Z.
'*           : All Rights Reserved
'* Date      : 26-2-2015
'* Version   : 1.0
'* Notes     : SOFTWARE VOOR HET BOEK.

Device 12F683           ; processor type 12F683

Xtal 10                 ; Kristal 10MHz

Config HS_OSC,_         ; Oscilator op 10MHz
        WDT_OFF,_       ; WatchDog Timer uit
        PWRTE_ON,_      ; Power-up Timer Enable aan
        MCLRE_OFF       ; Externe Master Reset Enable uit

All_Digital TRUE       ; Alle ingangen digitaal

Declare CCP1_Pin GPIO.2 ; PWM uitgang

Symbol FREQUENTIE = 38000 ; Frequentie 38KHz

Symbol VRIJGAVE = GPIO.0 ; Ingang vrijgave IR zender

Clear                  ; Wis geheugen

        ;543210        ; Hulpregel Poort IO
GPIO = %000000         ; Zet poort op nul
TRISIO = %000001      ; Poort I/O

;-----
; PROGRAMMA IR ZENDER LICHTSLUIS MET TSOP1738.
;-----

```

```

start:
  If VRIJGAVE = 1 Then
    HPWM 1,127,FREQUENTIE
    DelayMS 1
    HPWM 1,0 ,FREQUENTIE
    DelayMS 1
  Else
    HPWM 1,0,FREQUENTIE
  EndIf
GoTo start

End

```

Zoals je kan zien is de code vrij simpel voor de zender. De CONFIG en de andere instellingen zijn heel anders dan voor de grote controller. Voor de zender wordt een IR_LED gebruikt, deze led wordt met een pwm signaal aangestuurd, zoals je kan zien is dit een duty cycle signaal met een frequentie van 38KHz. Dat staat hier HPWM 1, 127, FREQUENTIE. De waarde 127 is de helft van 255, dus het signaal is net zo lang hoog als laag. Boven aan het programma staat onder SYMBOL FREQUENTIE = 38000, 38KHz dus. Het vrijgave signaal komt van het ontvanger gedeelte, dit is nodig om de TSOP1738 (die in de ontvanger zit) goed te laten werken. Er komt om de seconden een puls van de ontvanger die de zender even uit zet. Die puls gaat niet via IR, maar is via bedrading uitgevoerd.

We gaan nu het programma voor de ontvanger maken.

```

'* Name      : BOEK IR ONTVANGER VOOR LICHTSLUIS.
'* Author    : H van Zwieten.
'* Notice    : Copyright (c) 2015 H.v.Z.
'*           : All Rights Reserved
'* Date      : 26-2-2015
'* Version   : 1.0
'* Notes     : SOFTWARE VOOR HET BOEK.

Device 12F675           ; processor type 12F675

Xtal 10                 ; Kristal 10MHz

Config HS_OSC,_         ; Oscilator op 10MHz
        WDT_OFF,_       ; WatchDog Timer uit
        PWRTE_ON,_     ; Power-up Timer Enable aan
        MCLRE_OFF      ; Externe Master Reset Enable uit

All_Digital TRUE       ; Alle ingangen digitaal

Declare Adin_Res = 8    ; resolutie 8 bits
Declare Adin_Tad = frc  ; set RC osc
Declare Adin_Stime = 50 ; sample tijd 5

Symbol VRIJGAVE = GPIO.2 ; Ingang vrijgave IR zender

Symbol LED = GPIO.1     ; Led in optocoupler

Dim WAARDE As Byte     ; Variabele waarde

```

```

Dim TELLER As Byte           ; Variabele waarde

Clear                         ; Wis geheugen

                               ;543210           ; Hulpregel Poort IO
GPIO = %000000               ; Zet poort op nul
TRISIO = %000001             ; Poort I/O

                               ;76543210         ; hulpregel control register
ADCON0 = %00000001          ; A/D control register

;-----
; PROGRAMMA IR ONTVANGER LICHTSLUIS MET TSOP1738.
;-----

TELLER = 0

START:
  VRIJGAVE = 1

  WAARDE = ADIn 0

  If WAARDE > 175 Then
    LED = 1
    DelayMS 100
  Else
    LED = 0
  EndIf

  If WAARDE > 175 Then
    TELLER = TELLER + 1
    If TELLER > 10 Then
      GoSub RESET_VRIJGAVE
    EndIf
  EndIf

  If WAARDE < 150 Then
    TELLER = 0
  EndIf
GoTo START

RESET_VRIJGAVE:
  VRIJGAVE = 0
  TELLER = 0
  DelayMS 1000
  Return

End

```

Er wordt een analoge ingang gebruikt om de TSOP1738 uit te lezen. De 38KHz puls komt binnen op de TSOP1738, en van dat signaal wordt door een opamp schakeling een 0 tot 5VDC signaal gemaakt. Het signaal komt overeen met een spanning van +/- 2.5 volt, en dat komt overeen met een digitale waarde van +/- 127. Hoe komen we hier aan: Als we uitgaan van een spanning van 5VDC, (want daar werken de controllers op) kan je zeggen dat een acht bits waarde overeen komt met (5 : 255 = 0.0196). Dat is dus 0.0196 volt per stapje van de analoge ingang. We hebben uit de TSOP1738 een spanning van +/- 2.5 volt, dat is dus een digitale waarde van 127 (2.5 : 0.0196 = 127).

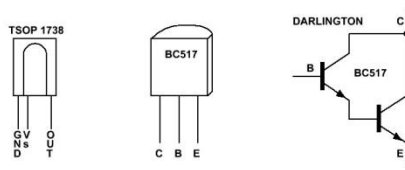
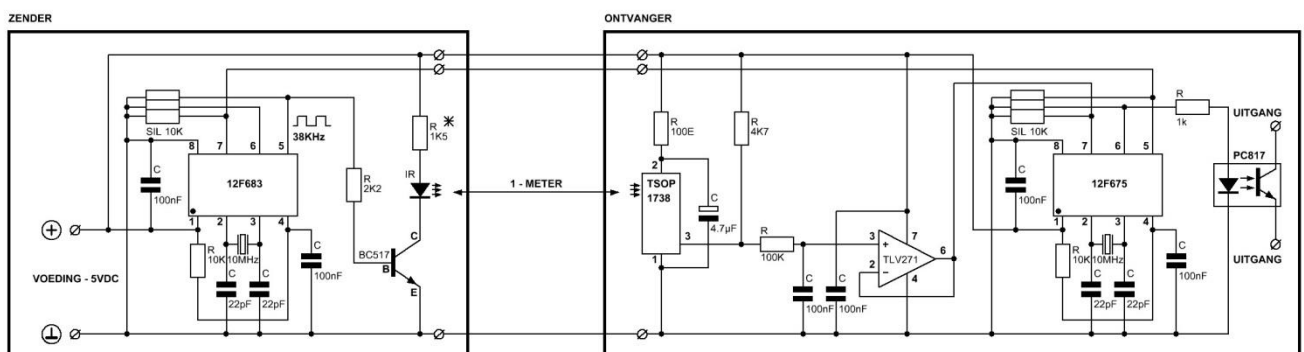
We zien boven in het programma `VRIJGAVE = 1` die is normaal dus altijd hoog, daar onder wordt de analoge waarde in een variabele gezet. Dan komt de vergelijking `IF WAARDE > 175` en daar gaat het om, we hebben gezien dat zolang het 38KHz signaal op de TSOP1738 staat dat de spanning 2.5 volt is. Als er nu bijv. iemand tussen de IR straal door loopt, dan wordt de 38KHz puls onderbroken en wordt de uitgang hoog. Die is dus nu $5 : 0.0196 = +/- 255$, die waarde is dus nu groter dan 175. Er gaat nu een teller lopen, en als de teller waarde `> 10` is dan wordt er naar `RESET_VRIJGAVE` gesprongen. Daar wordt `VRIJGAVE` op nul gezet, de `TELLER` wordt op nul gezet, en na één seconden wordt er weer naar `START` gesprongen. Zolang er dus iemand tussen de straal staat, blijft de schakeling resetten. Dit is gedaan omdat als de TSOP1738 te lang geen puls krijgt de TSOP1738 anders in verzadiging gaat en vast loopt.

Als er niemand tussen de straal staat wordt de teller automatisch gereset `IF WAARDE < 150 THEN TELLER = 0`. Tot zover de basic programma's

Als de lichtsluis onderbroken wordt dan schakelt de optocoupler de ingang op het controllerboard, en zal de teller (`TELLER_4` IN HET PROFILAB PROGRAMMA) met één ophogen.

Hier onder het schema van de lichtsluis.

DE TE OVERBRUGGEN AFSTAND HANGT AF VAN DE WAARDE VAN DE WEERSTAND
 GROTERE WAARDE WORDT EEN KORTERE AFSTAND
 BIJ BIJV. 4K7 WORDT DE AFSTAND +/- 50 CENTIMETER
 BIJ BIJV. 5K7 WORDT DE AFSTAND +/- 25 CENTIMETER



De infrarood led wordt door een BC517 aangestuurd. De te overbruggen afstand van de lichtsluis wordt bepaald door de stroom die door de infrarood led gaat, de waarde van de weerstand met de * aanduiding heeft een waarde van 1K5. Met deze waarde kan je een afstand van +/- één meter overbruggen. Als de waarde van de weerstand kleiner wordt, dan kan men een grotere afstand overbruggen. In de datasheet van de TSOP1738 kan weer info gevonden worden, die kan je op het net vinden. Tot zover dit voorbeeld, ik hoop dat dit een leuke toevoeging is.

6.5: Het serieel besturen van een servo

We gaan nu serieel een modelbouw servo aansturen via Profilab. Dit gaan we doen met de potmeter op de frontplaat. We gaan weer beginnen met het basic programma.

```
'* Name      : BOEK HET SERIEEL BESTUREN VAN EEN SERVO.
'* Author    : H van Zwieten.
'* Notice    : Copyright (c) 2015 H.v.Z.
'*          : All Rights Reserved
'* Date      : 27-2-2015
'* Version   : 1.0
'* Notes     : SOFTWARE VOOR HET BOEK.

Device 16F887                ; Processor type

Xtal 10                      ; Cristal 10Mhz

Asm                          ; Config settings
CONFIG_REQ
__CONFIG __CONFIG1, HS_OSC & WDT_OFF & DEBUG_OFF & FCMEN_OFF & LVP_OFF &
IESO_OFF & BOR_OFF & CPD_OFF & CP_OFF & MCLRE_OFF & PWRTE_ON
__CONFIG __CONFIG2, WRT_OFF & BOR40V
EndAsm

All_Digital true           ; Alle poorten digitaal

Declare Adin_Res = 8       ; Resolutie 8 bits
Declare Adin_Tad = frc     ; set RC osc
Declare Adin_Stime = 50    ; sample tijd 5

Declare CCP1_Pin PORTC.2   ; HPWM 1 op poort C.2
Declare CCP2_Pin PORTC.1   ; HPWM 2 op poort C.1

Declare Serial_Baud 9600   ; Baudrate 9600

Declare Rsin_Pin  PORTB.0   ; Data in
Declare Rsout_Pin PORTB.1   ; Data uit

Symbol SERVO_MOT = PORTC.0 ; Servo uitgang

Dim DATA_IN As Byte      ; Data serieele poort

Dim POSITIE As Word       ; Variabele waarde positie

Clear                     ; Wis geheugen

        ;543210           ; Hulpregel poort A
PORTA = %000000          ; Maak poort A laag
TRISA = %111111         ; Poort_A I/O

        ;543210           ; Hulpregel poort B
PORTB = %000000          ; Maak poort B laag
TRISB = %000001         ; Poort_B I/O

        ;76543210        ; Hulpregel poort C
PORTC = %00000000        ; Maak poort C laag
TRISC = %00000000        ; Poort_C I/O
```

```

;76543210 ; Hulpregel poort D
PORTD = %00000000 ; Maak poort D laag
TRISD = %00000000 ; Poort_D I/O

;210 ; Hulpregel poort E
PORTE = %000 ; Maak poort E laag
TRISE = %000 ; Poort_E I/O

;76543210 ; Hulpregel analoog
ADCON0 = %00000001 ; ADCON0 register analoog

;543210 ; Hulpregel analoog poort_B
ANSELH = %000000 ; ANSEL register analoog
poort_B

;-----
; HIER ONDER DE CODE VOOR HET SERIEEL AANSTUREN VAN EEN MODELBOUW SERVO.
;-----

RUN:
  DATA_IN = RSIn
  POSITIE = 850 + (DATA_IN*7)
  Servo SERVO_MOT, POSITIE
  DelayMS 20
GoTo RUN

End

```

Zoals je ziet valt de code weer mee voor dit voorbeeld. De RSIN data wordt weer in de variabele DATA_IN gezet. De minimale positie van de servo is 850, daar wordt de DATA_IN X 7 bij opgeteld dus de max positie is $255 \times 7 = 1785 + 850 = 2635$. Met het commando SERVO kan dus heel simpel de servo bediend worden. Achter het commando SERVO komt de poort_pin, in dit geval is dat PORTC.0 die hebben we zo genoemd onder SYMBOL SERVO_MOT = PORTC.0. En daar achter komt de positie data, en in dit geval komt die waarde uit DATA_IN.

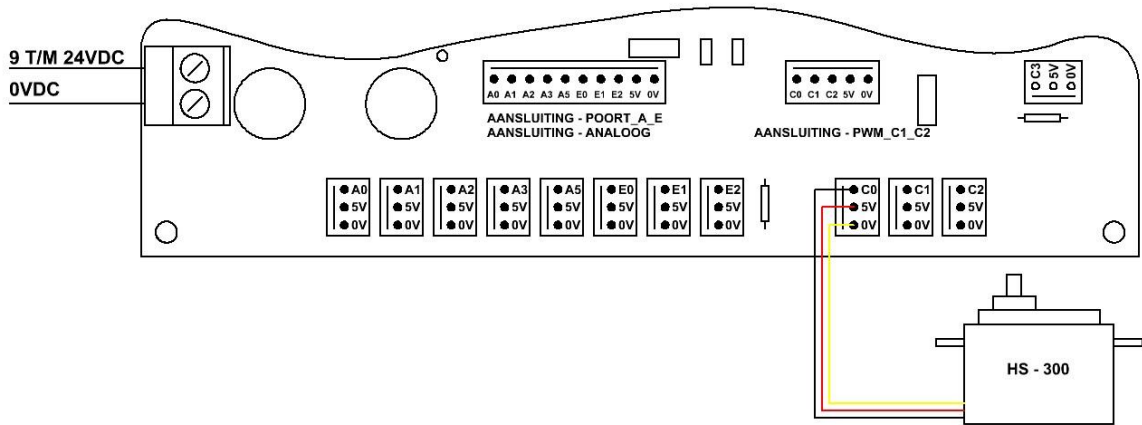
Nog even over de werking van een servo.

In de servo zit een motor met een vertraging, de uitgaande as van de servo is gekoppeld aan een interne potmeter. Deze potmeter is gekoppeld aan de interne elektronica van de servo. De inkomende pulsen en het potmeter signaal worden door de elektronica met elkaar vergeleken en sturen de motor aan als dat nodig is. De potmeter vormt dus een terugkoppeling naar de elektronica. De servo werkt met signalen tussen de één en twee milliseconden, tussen die waarde beweegt de servo naar links en rechts.

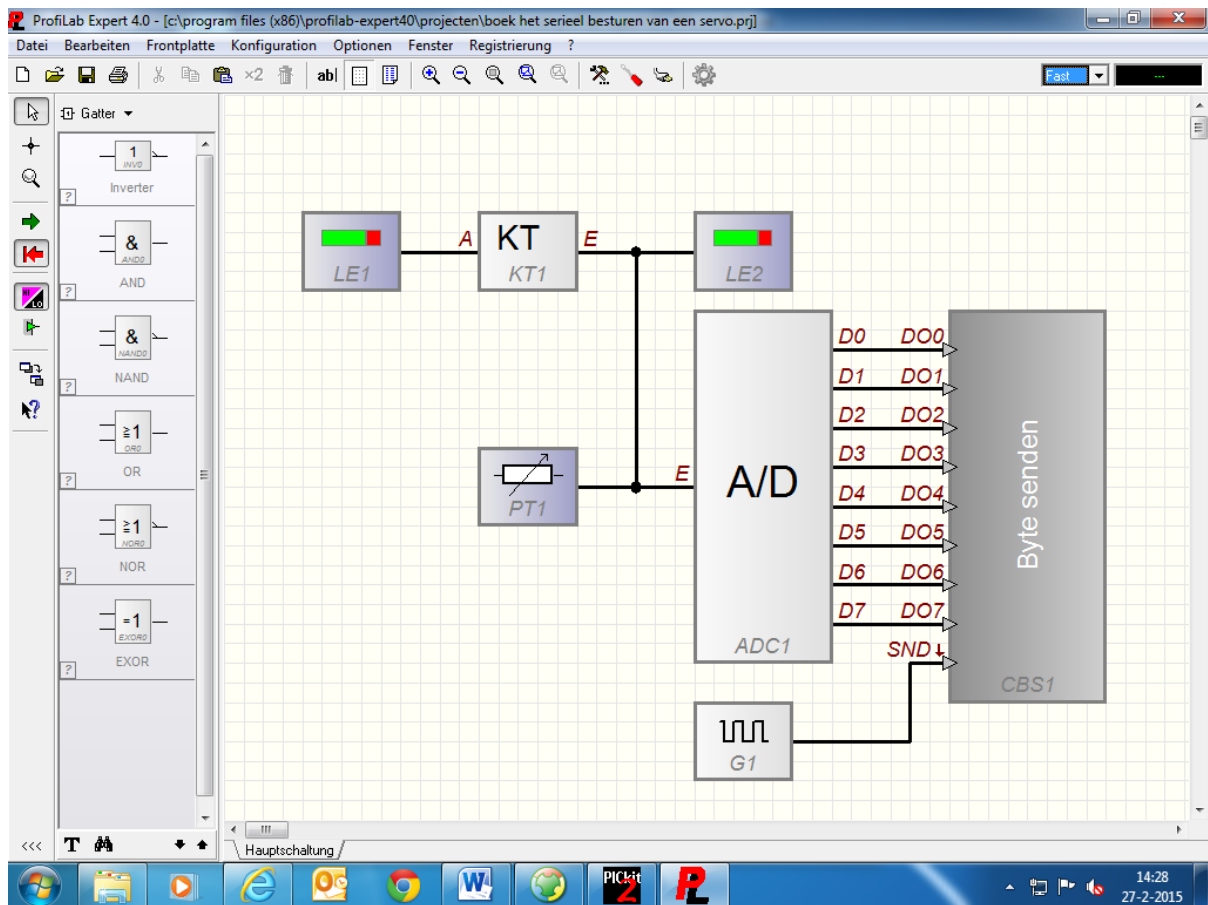
De gebruikte servo in het voorbeeld is een HITEC – HS – 300. De puls waarde van 850 tot 2635 is goed voor dit type servo, maar dat kan per merk en type variëren. Dus daar moet even mee geëxperimenteerd worden, Een waarde tussen de 1000 en 2000 zou altijd moeten werken, maar dan heb je maar een kleine uitslag op de uitgaande as van de servo.

De stekker van de servo kan rechtstreeks aangesloten worden op de Molex connector die op het controllerboard zit.

Zie aansluitschema op blz. 87.



Tot zover het basic programma. We gaan nu het Profilab programma maken.



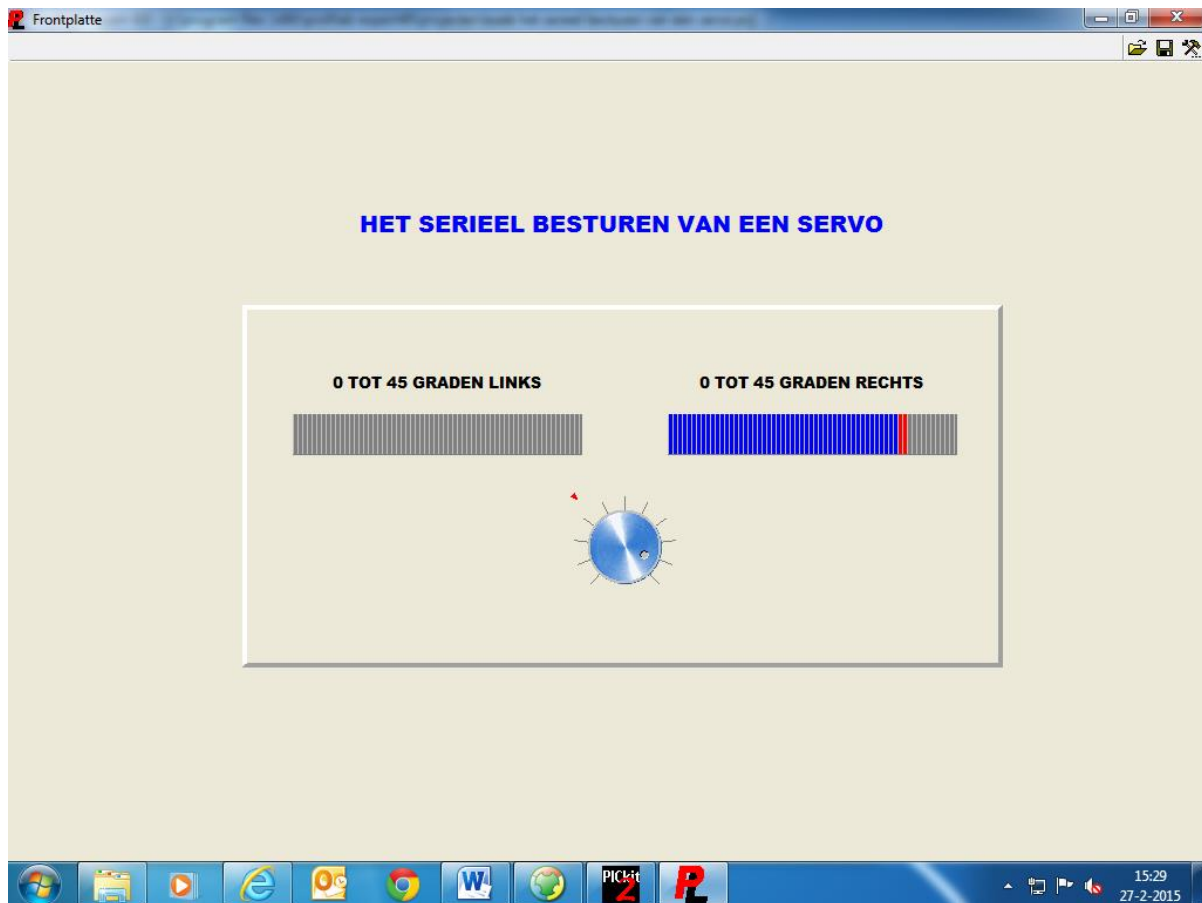
We slepen één potmeter, één puls geveer, één correctie module, twee led bars, één analoog/digitaal module, en één acht bit seriële module naar het werkblad.

De potmeter PT1 komt aan de ingang E van de analoog/digitaal module ADC1, aan de E ingang van de correctie module KT1, en aan de E ingang van de led bar LE2. De uitgang van de correctie module komt aan de E ingang van led bar E1.

De uitgangen D0 tot en met D7 van de analoog/digitaal module komen één op één aan DO0 tot en met DO7 van de seriële module CBS1. En de puls geveer G1 komt aan de SND ingang van de seriële module CBS1.

De waarde van de puls geveer G1 zetten we op honderd (100Hz). De waarde van de analoog/digitaal module stellen we in tussen 0 en 255. De waarde die in de seriële module komt te staan is weer COM5, 9600 BAUD, DATA 8, PARITY NONE, en STOP 1. De led bar LE1 stellen we in op links en de kleur tot 80% zetten we op blauw, en van 80 tot 100% zetten we de kleur op rood. De led bar E2 stellen we in op rechts en de kleur tot 80% zetten we op blauw, en van 80 tot 100% zetten we ook weer op rood. Dit kan je naar eigen inzicht instellen. Nu hebben we nog de correctie module KT1, als we die onder eigenschappen openen zie je twee regels staan waar je een waarde in kan geven. Op de bovenste regel zetten we bij ingang de waarde 0, en bij uitgang zetten we de waarde 127. Op de tweede regel zetten we bij de ingang waarde 127, en bij de uitgang zetten we de waarde 0. Op deze manier kan je de analoge waarde inverteren.

We gaan de frontplaat weer maken, de indeling heb ik zo gemaakt.



Zoals je kan zien heb ik weer een raam gebruikt, waar de twee led bars en de potmeter in staan. Er is ook weer wat tekst toegevoegd.

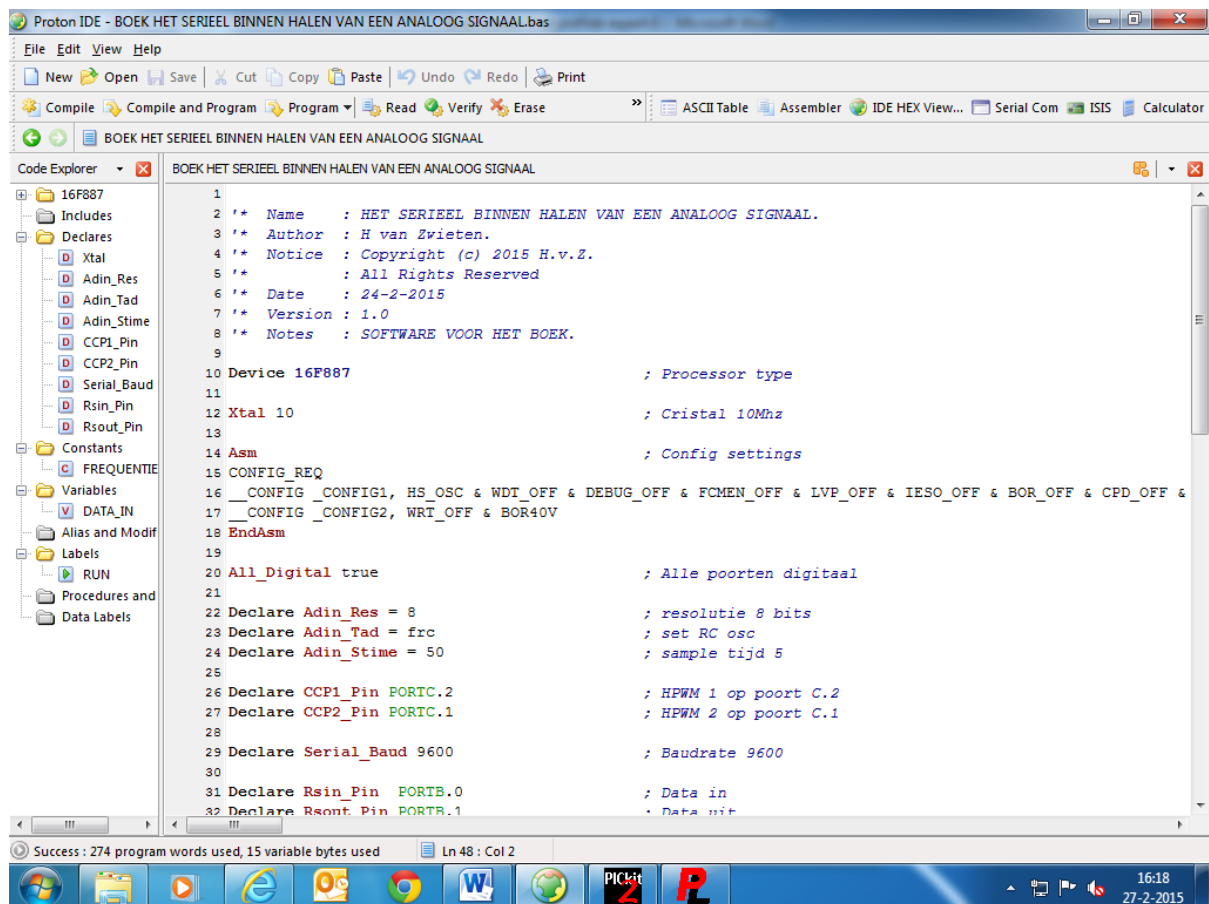
Als de potmeter in de midden stand staat dan zijn beide led bars blank, draai je de potmeter naar rechts dan loopt de rechter led bar van links naar rechts vol. Draai je de potmeter van uit de midden stand naar links, dan loopt de linker led bar van rechts naar links vol.

Zoals je kan zien aan de stand van de potmeter en aan de rechter led bar, staat de servo nu iets meer dan 80% naar rechts verdraaid. We gaan dit voorbeeld weer afsluiten.

6.6: Het serieel besturen van een dimmer

We gaan een dimmer besturen met Profilab, dit gaan we weer doen met een potmeter op de frontplaat. We kunnen daar het zelfde programma voor gebruiken als in het voorbeeld, **HET SERIEEL BINNEN HALEN VAN EEN ANALOOG SIGNAAL**.

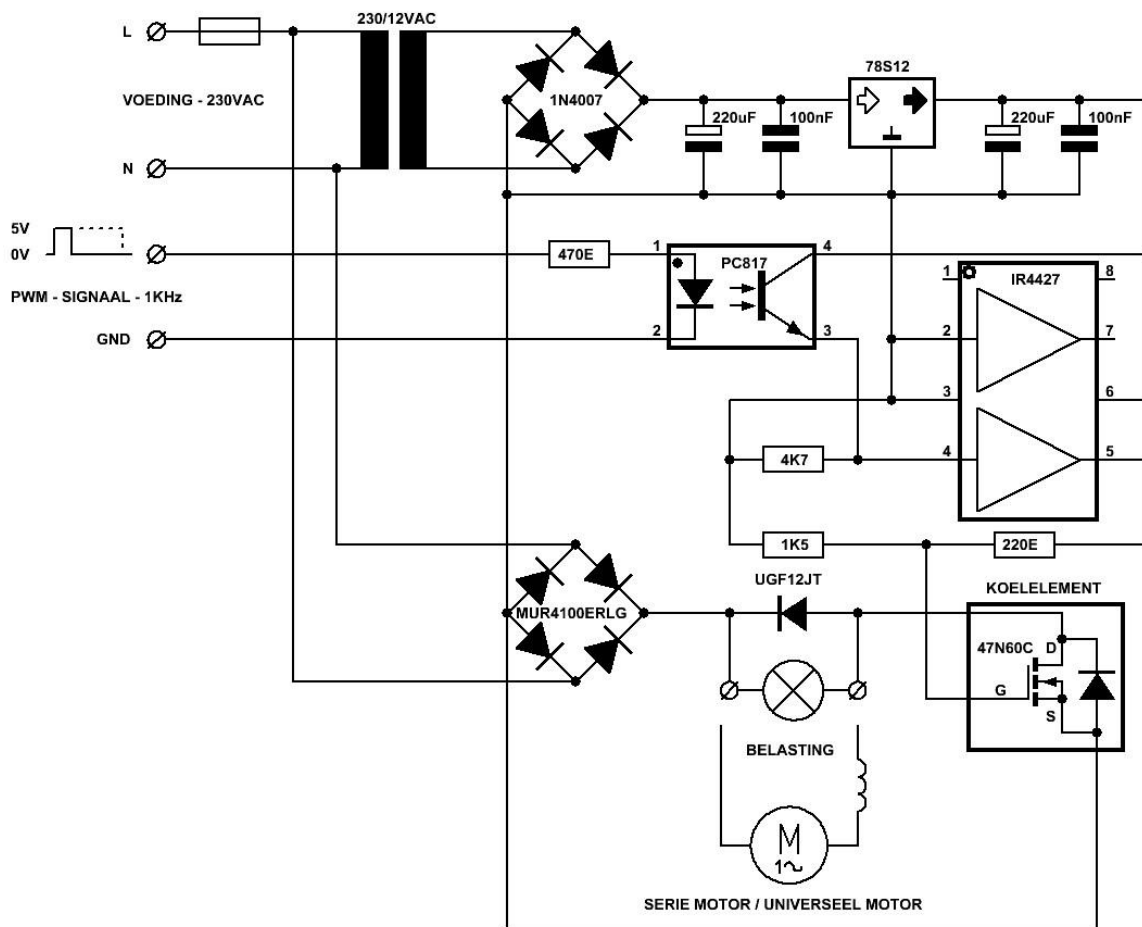
we gaan dus weer het pwm signaal gebruiken voor dit voorbeeld.



```
1
2  /* Name      : HET SERIEEL BINNEN HALEN VAN EEN ANALOOG SIGNAAL.
3  /* Author    : H van Zvieten.
4  /* Notice    : Copyright (c) 2015 H.v.Z.
5  /*          : All Rights Reserved
6  /* Date      : 24-2-2015
7  /* Version   : 1.0
8  /* Notes     : SOFTWARE VOOR HET BOEK.
9
10 Device 16F887                ; Processor type
11
12 Xtal 10                       ; Cristal 10Mhz
13
14 Asm                            ; Config settings
15 CONFIG_REQ
16 _CONFIG_CONFIG1, HS_OSC & WDT_OFF & DEBUG_OFF & FCMEN_OFF & LVP_OFF & IESO_OFF & BOR_OFF & CPD_OFF &
17 _CONFIG_CONFIG2, WRT_OFF & BOR40V
18 EndAsm
19
20 All_Digital true              ; Alle poorten digitaal
21
22 Declare Adin_Res = 8           ; resolutie 8 bits
23 Declare Adin_Tad = Frc        ; set RC osc
24 Declare Adin_Stime = 50       ; sample tijd 5
25
26 Declare CCP1_Pin PORTC.2      ; HPWM 1 op poort C.2
27 Declare CCP2_Pin PORTC.1      ; HPWM 2 op poort C.1
28
29 Declare Serial_Baud 9600      ; Baudrate 9600
30
31 Declare Rsin_Pin PORTB.0       ; Data in
32 Declare Rsout_Pin PORTB.1      ; Data uit
```

Ik ga het basic software voorbeeld niet meer plaatsen, want dat staat al in het boek, we gaan eerst het schema plaatsen en bespreken.

Het schema van de dimmer staat op Blz. 90.



Hier boven het schema van de dimmer. Zoals je kan zien is het geen standaard dimmer, de 230VAC net spanning wordt gelijk gericht en door een mosfet met een pwm frequentie van 1KHz geschakeld. En toch kan je met de dimmer zowel een lamp als een elektromotor regelen. De elektromotor moet wel een serie motor of een universeel motor zijn. Voor het besturingsgedeelte is een kleine trafo opgenomen van 230 / 12VAC / 1.5VA. Na gelijkrichting en afvlakking houden we daar een gelijkspanning van $12 \times 1.4 = 16.8\text{VDC}$ aan over, genoeg voor de 12V regelaar 78S12. De waarde 1.4 komt van wortel 2 = 1.4142135, ik houd voor het gemak 1.4 aan. Met de 12VDC uit de spanningsregelaar wordt de mosfet driver IR4427 gevoed. De driver voorziet de gate van de mosfet van spanning en stroom. Die stroom die er loopt is heel laag want we zeggen altijd dat een mosfet spanning gestuurd is, in tegenstelling tot een transistor. Met het inkomende pwm signaal wordt de optocoupler (PC817) bestuurd, en die voorziet op zijn beurt de drijver van een stuur signaal.

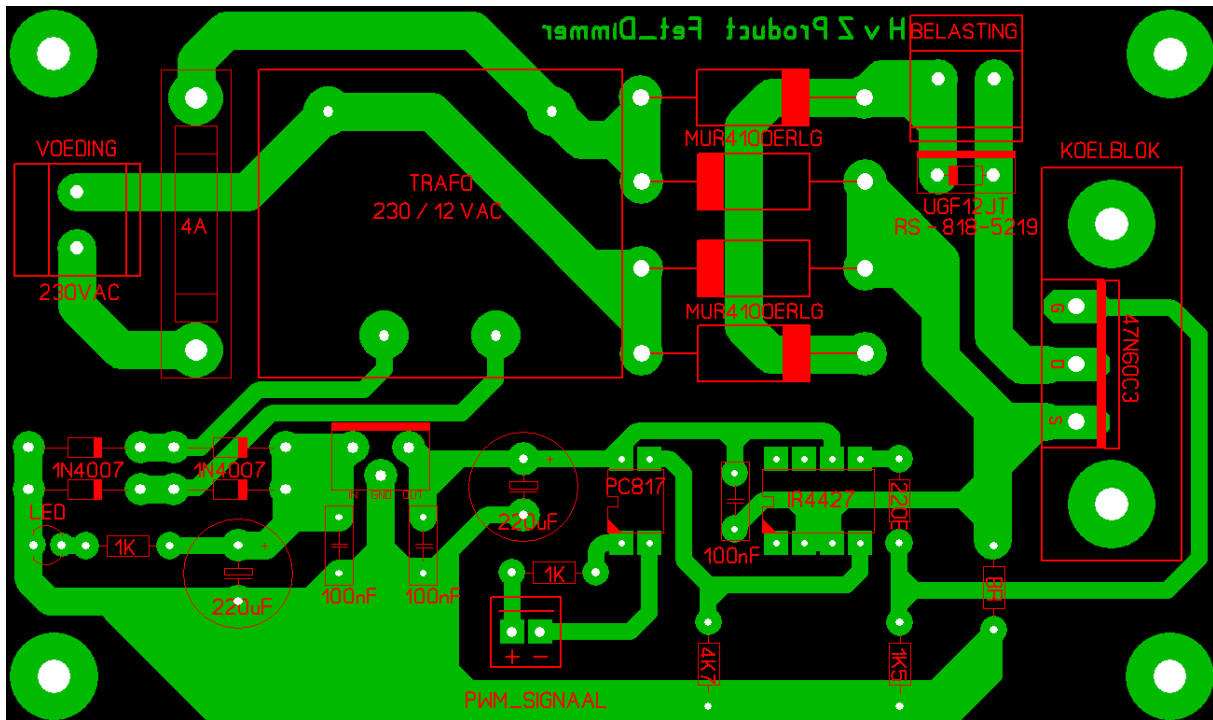
De mosfet is van het type 47N60C, deze mosfet mag 600VDC schakelen met een stroom van max 47A. De R_{DSon} (drain – source overgang) is maar 70 milli ohm, dat is dus heel gunstig. Op deze manier verstoekt je weinig warmte in de mosfet, en heb je dus maar een klein koellichaam nodig.

Voor inductieve belastingen is er een diode van het type UGF12JT opgenomen, Voor het gelijkrichten van de hoofdstroom wordt de MUR4100ERJG gebruikt. Die diodes zijn 4A, dus de dimmer mag max +/- 900 watt schakelen. De hele schakeling is beveiligd door een zekering van 4A.

Door de PC817 is het stuurgedeelte galvanisch van de netspanning gescheiden, er kan dus nooit een spanning van 230VAC op het controllerboard komen.

Maar Let er wel op dat de schakeling een gevaarlijke spanning voert, dus pas daar mee op!

Er is ook een print voor de dimmer gemaakt, zie hieronder.



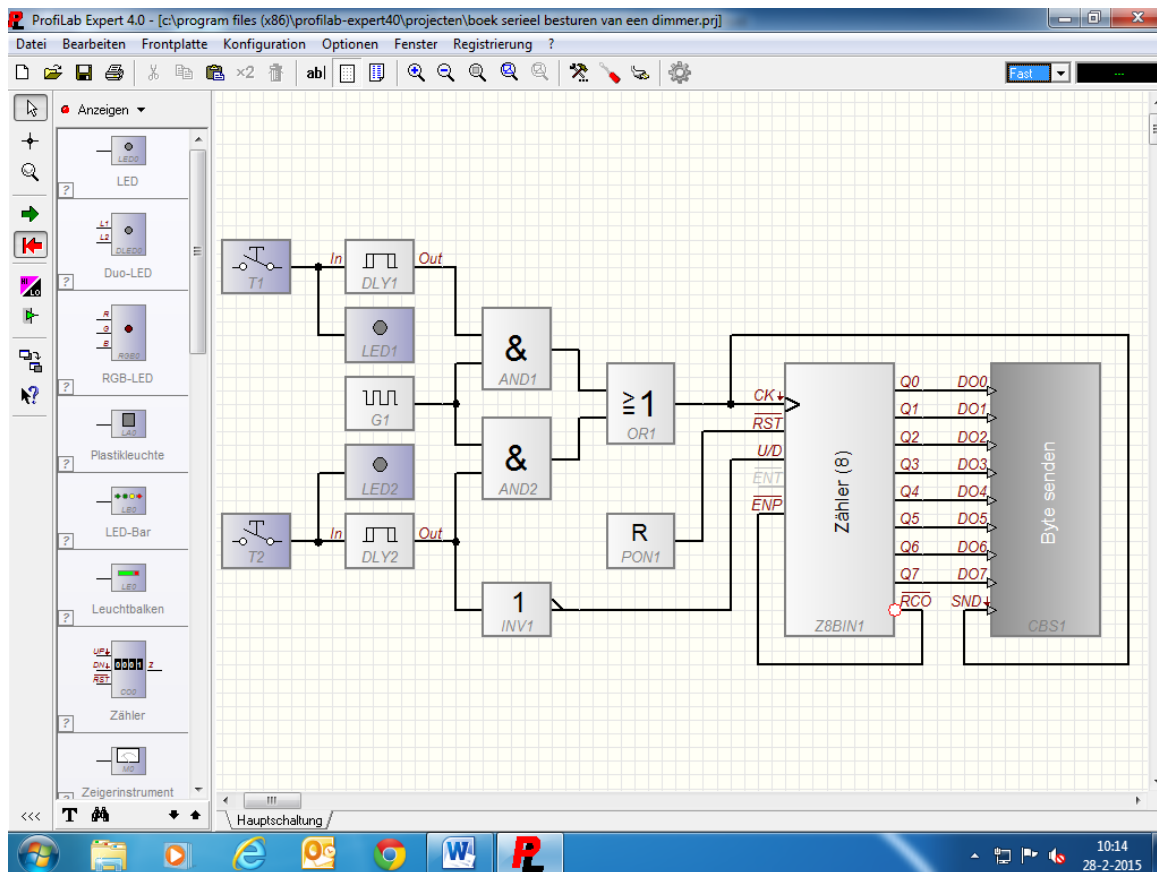
De print heeft een formaat van 110 X 65mm. Dit kan ook weer op gaatjes print gemaakt worden, als je dat doet let wel op de afstand tussen de delen die 230VAC voeren.

Zelf frees ik mijn printen tegenwoordig dat gaat perfect geen geknoei meer met ets middelen, en je kan een paar slagen overslaan. Niet meer eerst printen op een sheet, en niet meer belichten.

Voor de isolatie spuit ik de printen een keer of vier in de plastic spray, ik gebruik daar PASTIEK_70 voor. Dat is te koop bij diverse elektronica zaken, of via winkels op internet. Het is prima spul, en het isoleert goed. Het is alleen een ramp als je een component moet vervangen, dat is dan weer een nadeel.

We gaan nu het Profilab programma maken.

We gaan dit keer geen potmeter gebruiken om de dimmer te bedienen, maar drukknoppen. Dus we krijgen twee drukknoppen één voor DIM_OP, en één voor DIM_NEER.



We slepen Twee drukknoppen, twee leds, twee vertraging modules, één puls geveer, twee and poorten, één inverter, één or poort, één reset module, één acht bits teller module, en één acht bit seriële module naar het werkblad. Drukknop T1 komt aan vertraging module DLY1 en aan LED1, en drukknop T2 komt aan vertraging module DLY2 en aan LED2. De uitgangen van vertraging module DLY1 en DLY2 komen aan and poort AND1 en AND2. Twee ingangen van AND1 en AND2 zijn met elkaar verbonden, en daar komt puls geveer G1 aan. Op de uitgang van vertraging module DLY2 is ook nog inverter INV1 aan gesloten. De uitgang van de inverter INV1 komt aan de U/D ingang van teller Z8BIN1, en de reset module PON1 komt aan de RST ingang van teller Z8BIN1. De twee uitgangen van de and poorten AND1 en AND2 komen aan de ingangen van de or poort OR1, de uitgang van de or poort komt aan de CK ingang van de teller Z8BIN1, en aan de SND ingang van de seriële module CBS1. De uitgangen van de teller Z8BIN1 (Q0 tot en met Q7) komen één op één aan de ingangen van de seriële module CBS1. De uitgang RCO van teller Z8BIN1 komt aan de ingang ENP van de teller, ook wordt de RCO pin weer geïnverteerd van de teller door er even op te klikken.

De delay module DLY1 en DLY2 achter de drukknoppen T1 en T2 zorgen voor een kleine vertraging, dit is gedaan voor een zachte regeling van de dimmer. De waarde van de vertraging modules zetten we op drie, dit komt over een met een vertraging van 150 milliseconden.

De drukknop T1 is voor de functie DIM_OP, en T2 is voor de functie DIM_NEER.

De waarde van de puls geveer staat op honderd, 100Hz dus.

De waarde die in de seriële module komt te staan is weer COM5, 9600 BAUD, DATA 8, PARITY NONE, en STOP 1.

We gaan de frontplaat weer maken.



Zoals je kan zien heb ik de frontplaat zo ingedeeld, er is weer een raam gebruikt, er staat tekst op de drukknoppen en op de frontplaat. Met de drukknop DIM_OP gaat de lamp feller branden, en met de drukknop DIM_NEER wordt de lamp gedimd. Je kan de drukknoppen gewoon ingedrukt houden, als de teller een waarde van 255 of 0 heeft gehaald stopt de teller van zelf.

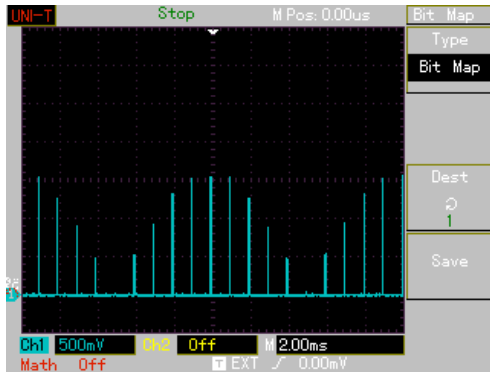
De aansluiting van de dimmer is als volgt. De ingang van de dimmer wordt aangesloten op PORTC.1 van het controllerboard, het schema daarvoor staat onder: **Het serieel uitsturen van een analog signaal.**

We sluiten dus de led (die in de optocoupler van de PC817 zit) op de zelfde manier aan zoals de led in het boven genoemde voorbeeld, op deze manier wordt de dimmer bediend.

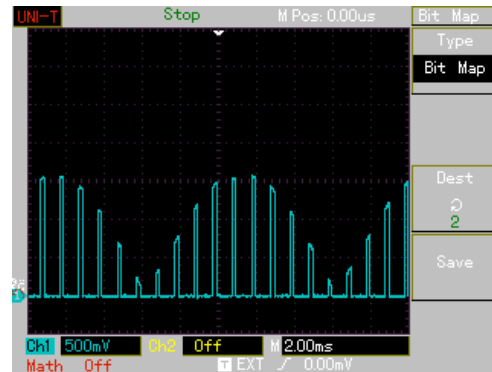
De led in de optocoupler geeft het pwm signaal door aan de interne transistor van de PC817, die op zijn beurt de driver voor de mosfet schakelt.

Ik ga nog wat scoopbeelden laten zien van het uitgang signaal van de dimmer. Dit is dus het gelijk gerichte 230VDC signaal wat op de lamp komt te staan.

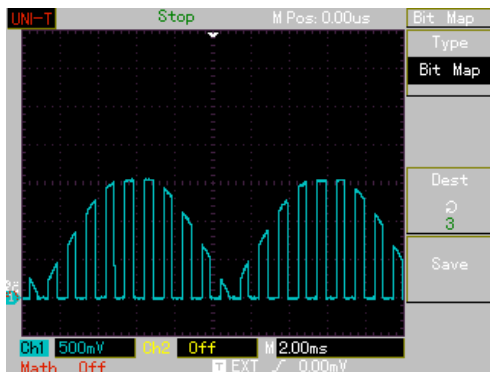
Zie Blz. 94.



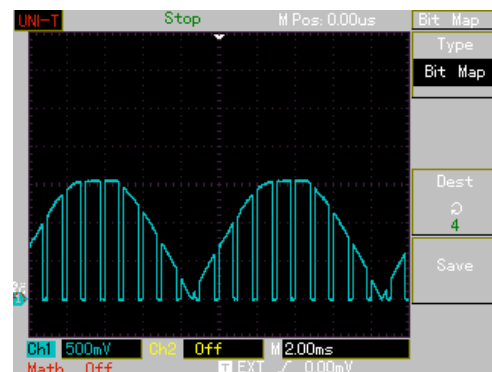
Hier boven staat de dimmer op 10%.



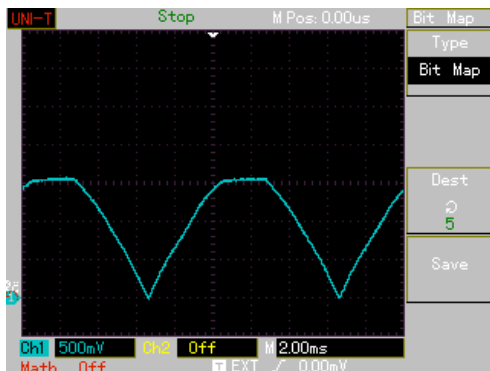
Hier boven staat de dimmer op 25%.



Hier boven staat de dimmer op 50%.



Hier boven staat de dimmer op 75%.



En hier staat de dimmer helemaal open gestuurd.

Op de beelden is de gelijkgerichte sinus te zien, hier is ook te zien dat het negatieve gedeelte van de sinus omgeklapt is naar boven. Dit signaal is dus ook van 50 naar 100Hz gegaan. Op de bovenste beelden is te zien dat daar het pwm signaal aan toegevoegd is. Er wordt dus in de sinus gehakt zoals dat genoemd wordt. Tot zover dit voorbeeld.

6.7: Het serieel besturen van een elektromotor

We gaan nu een elektromotor serieel besturen met Profilab. We gaan weer beginnen met het basic programma. Ik heb bij dit programma wat verklarende tekst gezet, omdat dit een langer programma is dan bij de eerdere voorbeelden.

Er wordt in dit voorbeeld ook gebruik gemaakt van een display, dat is aangesloten op het controllerboard. Het display zit op de PORTD.2, D3, en PORTD.4 tot en met D.7. Het display is van Intech type ITM1602K1 – 04. Dit is een display met twee regels en twee maal 16 karakters, de displays zijn ook te koop met vier regels en meer karakters.

Hier onder het basic programma.

```
'* Name      : BOEK HET SERIEEL BESTUREN VAN EEN ELEKTROMOTOR.
'* Author    : H van Zwieten.
'* Notice    : Copyright (c) 2015 H.v.Z.
'*          : All Rights Reserved
'* Date      : 1-3-2015
'* Version   : 1.0
'* Notes     : SOFTWARE VOOR HET BOEK.

Device 16F887                ; Processor type

Xtal 10                      ; Cristal 10Mhz

Asm                        ; Config settings
CONFIG_REQ
__CONFIG __CONFIG1, HS_OSC & WDT_OFF & DEBUG_OFF & FCMEN_OFF & LVP_OFF &
IESO_OFF & BOR_OFF & CPD_OFF & CP_OFF & MCLRE_ON & PWRTE_ON
__CONFIG __CONFIG2, WRT_OFF & BOR40V
EndAsm

All_Digital true          ; Alle poorten digitaal

Declare LCD_RSPin PORTD.2  ; Reset display poort D.2
Declare LCD_ENPin PORTD.3  ; Enable display poort D.3
Declare LCD_DTPin PORTD.4  ; Data display PORTD.4 t/m D.7

Declare Rsout_Pin PORTB.1   ; Data in
Declare Rsin_Pin PORTB.0    ; Data uit

Declare Serial_Baud 9600    ; Baudrate 9600

Declare CCP1_Pin PORTC.2    ; PWM poort C.2

Symbol FREQUENTIE = 4000    ; Frequentie 4KHz

Symbol LINKS = PORTC.0      ; Motor links
Symbol RECHTS = PORTC.1     ; Motor rechts
Dim DATA_COM_IN As Byte   ; Data serieel in
Dim DATA_COM_UIT As Byte ; Data serieel uit

Dim TOEREN As Byte        ; Variabele voor pwm waarde

Dim PWM_VAR As Byte      ; Variabele voor pwm percentage
```

```

Clear                                ; Wis geheugen

Cls                                    ; Wis display

DelayMS 500                            ; Pauze 0.5 sec

                                ;543210          ; Hulpregel poort A
PORTA = %000000                        ; Maak poort A laag
TRISA = %000010                          ; Poort_A I/O

                                ;543210          ; Hulpregel poort B
PORTB = %000000                        ; Maak poort B laag
TRISB = %000001                          ; Poort_B I/O

                                ;76543210        ; Hulpregel poort C
PORTC = %00000000                      ; Maak poort C laag
TRISC = %00000000                      ; Poort_C I/O

                                ;76543210        ; Hulpregel poort D
PORTD = %00000000                      ; Maak poort D laag
TRISD = %00000000                      ; Poort_D I/O

                                ;210             ; Hulpregel poort E
PORTE = %000                            ; Maak poort E laag
TRISE = %000                            ; Poort_E I/O

                                ;76543210        ; Hulpregel analoog
ADCON0 = %00000001                     ; ADCON0 register analoog

                                ;543210          ; Hulpregel analoog
ANSELH = %000000                       ; ANSEL register analoog

;-----
; HIER ONDER DE CODE VOOR HET SERIEEL BESTUREN VAN EEN ELEKTROMOTOR.
;-----

TOEREN = 0                             ; Zet variabele toeren op nul
RECHTS = 0                             ; Zet poort C.0 op nul
LINKS = 0                              ; Zet poort C.1 op nul
PWM_VAR = 0                            ; Zet variabele pwm op nul

MOTOR_IN_BEDRIJF:                    ; Start loop motor in bedrijf
  Print At 1,4,"PWM"                    ; Zet text op display
  Print At 1,12,"%"                     ; Zet text op display
  Print At 2,1,"  MOTOR_L__R  "        ; Zet text op display

```



```

If LINKS = 1 Then ; Als waar dan
  Print At 2,1," MOTOR_L_OM " ; Zet text op display
EndIf ; Einde als

If RECHTS = 1 Then ; Als waar adn
  Print At 2,1," MOTOR_R_OM " ; Zet text op display
EndIf ; Einde als

DATA_COM_IN = RSIn ; Zet data in variabele

HPWM 1,TOEREN,FREQUENTIE ; Set pwm signaal

PWM_VAR = TOEREN / 2.54 ; Bereken pwm signaal

If DATA_COM_IN = 10 Then ; Als data is waar dan
  Print At 1,8,Dec,PWM_VAR, " " ; Zet text op display
  TOEREN = TOEREN + 1 ; Op toeren van motor
  If TOEREN > 253 Then TOEREN = 254 ; Blokkeren toerental
  RSOut PWM_VAR ; Verstuur pwm signaal
  DelayMS 1 ; Pauze
EndIf ; Einde als

If DATA_COM_IN = 20 Then ; Als data is waar dan
  Print At 1,8,Dec, PWM_VAR, " " ; Zet text op display
  TOEREN = TOEREN - 1 ; Af toeren van motor
  If TOEREN < 2 Then TOEREN = 1 ; Blokkeren toerental
  RSOut PWM_VAR ; Verstuur pwm signaal
  DelayMS 1 ; Pauze
EndIf ; Einde als

If DATA_COM_IN = 30 Then ; Als data is waar dan
  Print At 2,1," MOTOR_L_OM " ; Zet text op display
  LINKS = 1 ; Poort C.1 is hoog
  RECHTS = 0 ; Poort C.0 is laag
EndIf ; Einde als

If DATA_COM_IN = 40 Then ; Als data is waar dan
  Print At 2,1," MOTOR_R_OM " ; Zet text op display
  RECHTS = 1 ; Poort C.0 is hoog
  LINKS = 0 ; Poort C.1 is laag
EndIf ; Einde als

If DATA_COM_IN = 50 Then ; Als data is waar dan
  GoTo WIS_DISPLAY ; Ga naar wis display
EndIf ; Einde als
GoTo MOTOR_IN_BEDRIJF ; Naar begin motor in bedrijf

```

```

NOODSTOP:                                     ; Start loop noodstop
    HPWM 1, TOEREN, FREQUENTIE                ; Set pwm signaal

    PWM_VAR = TOEREN / 2.54                   ; Bereken pwm signaal

    TOEREN = TOEREN - 1                       ; Af Toeren van motor
    If TOEREN < 2 Then TOEREN = 1           ; Blokkeren toerental
    RSOut PWM_VAR                             ; Verstuur pwm signaal
    DelayMS 1                                 ; Pauze

    If TOEREN = 1 Then                         ; Als waar dan
        Print At 1,1, "    RESET    "        ; Zet text op display
        Print At 2,1, "    NOODSTOP  "      ; Zet text op display
        GoTo WIS_DISPLAY_1                   ; Ga naar wisdisplay 1
    EndIf                                     ; Einde als
GoTo NOODSTOP                                 ; Naar begin noodstop

WIS_DISPLAY:                                  ; Start loop wis display
    Cls                                       ; Wis display
    DelayMS 100                               ; Pauze
    GoTo NOODSTOP                             ; ga naar loop noodstop

WIS_DISPLAY_1:                                ; Start loop wis display 1
    DATA_COM_IN = RSIn                       ; Zet data in variabele

    Cls                                       ; Wis display
    DelayMS 100                               ; Pauze

    If DATA_COM_IN = 70 Then                 ; Als data is waar dan
        GoTo MOTOR_IN_BEDRIJF               ; Ga naar loop motor in bedrijf
    EndIf                                     ; Einde als
GoTo WIS_DISPLAY_1                           ; Ga naar begin wis display 1

End                                           ; Einde programma

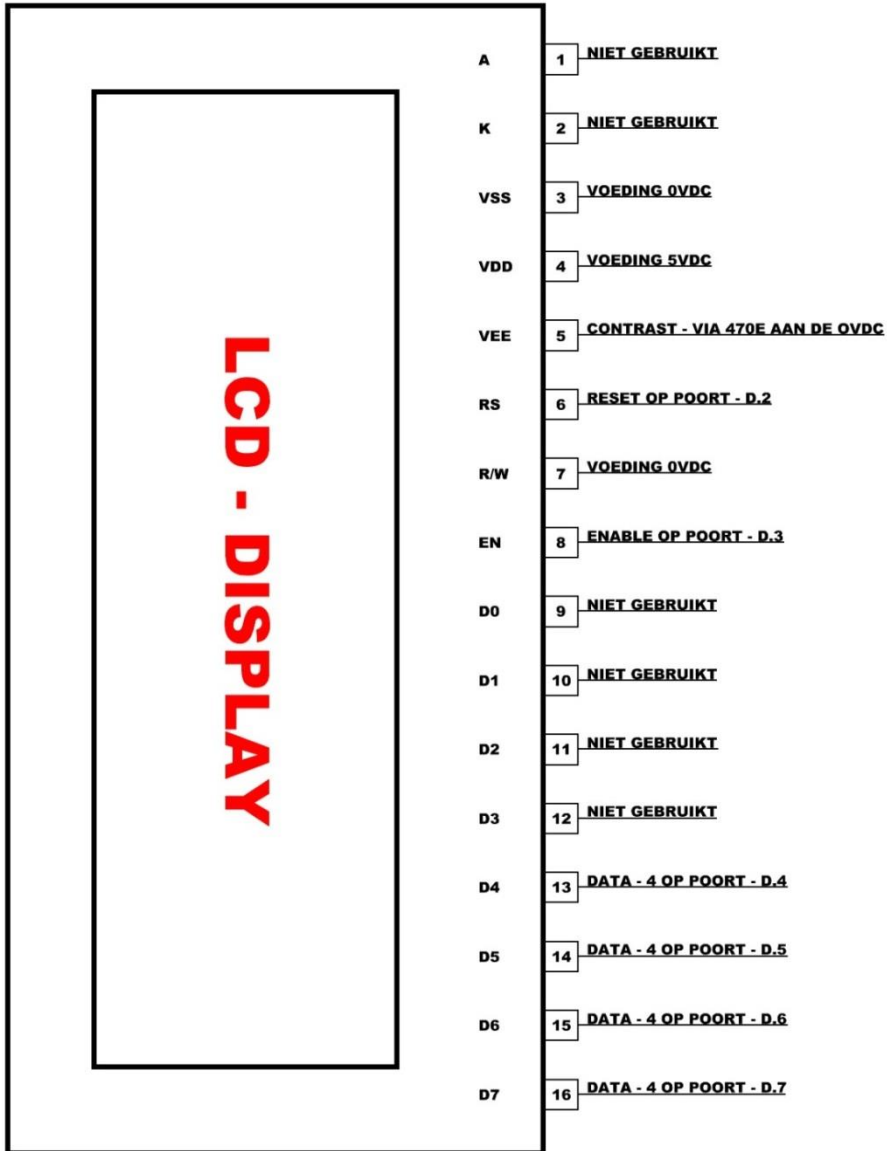
```

Tot zover het basic programma ik ga het programma niet verder toelichten, bekijk het eens rustig en probeer te begrijpen wat er gebeurt.

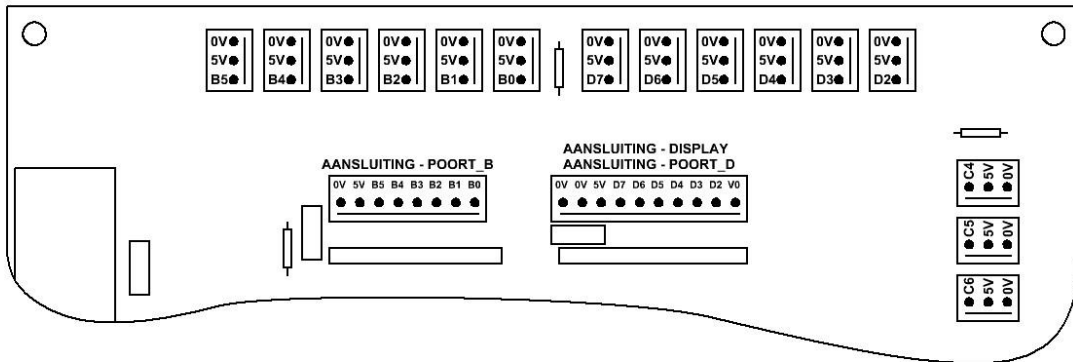
Met de tekst achter het programma moet je er wel uitkomen denk ik.

Nog even wat over het display, het commando PRINT zorgt ervoor dat er data naar het display gestuurd wordt. Dit gebeurt via de parallele methode, er zijn ook seriële displays te koop en displays die via I2C werken. Deze displays kunnen ook aangestuurd worden door de controller.

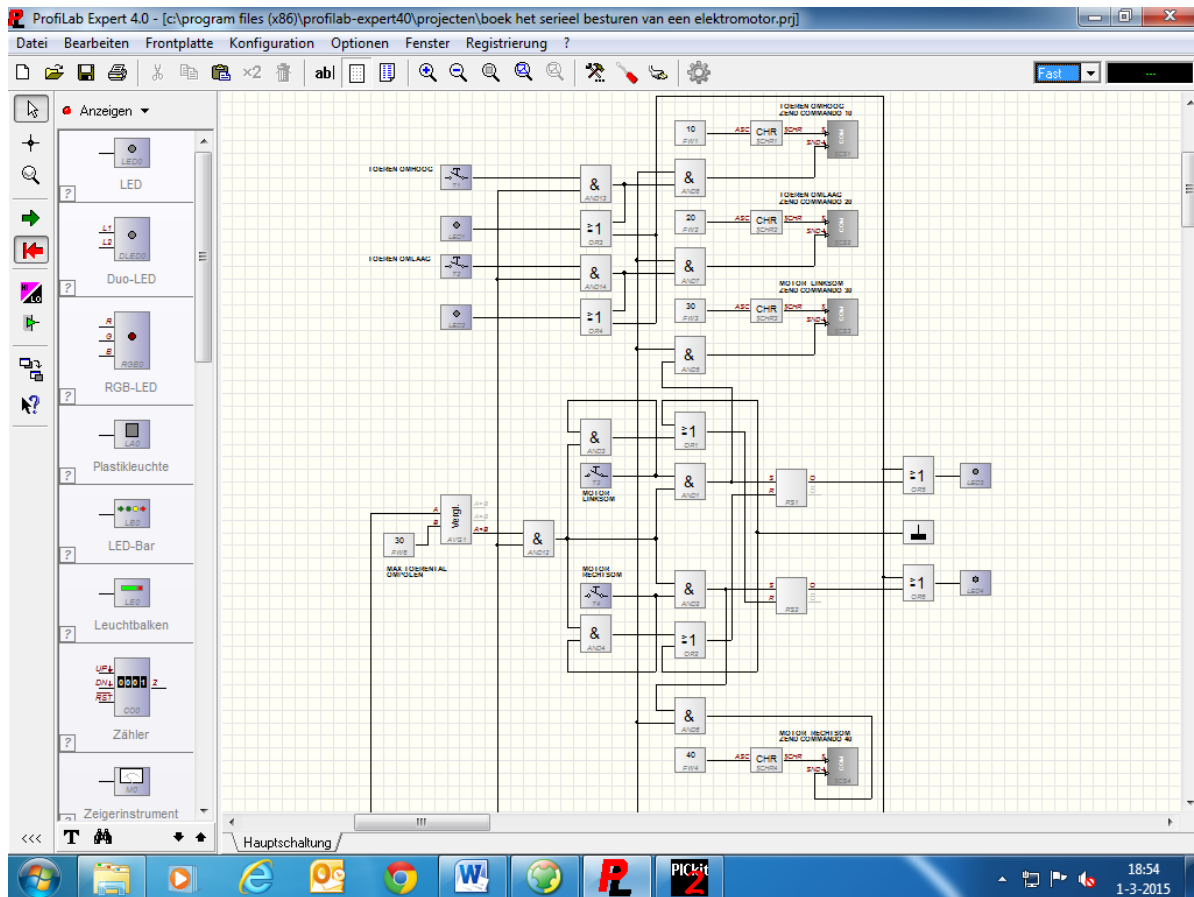
Op Blz. 99 zie je de aansluitingen voor het intech display. Bij de pinnen van het display staat op welke poort_pin het display aangesloten moet worden op de controller.



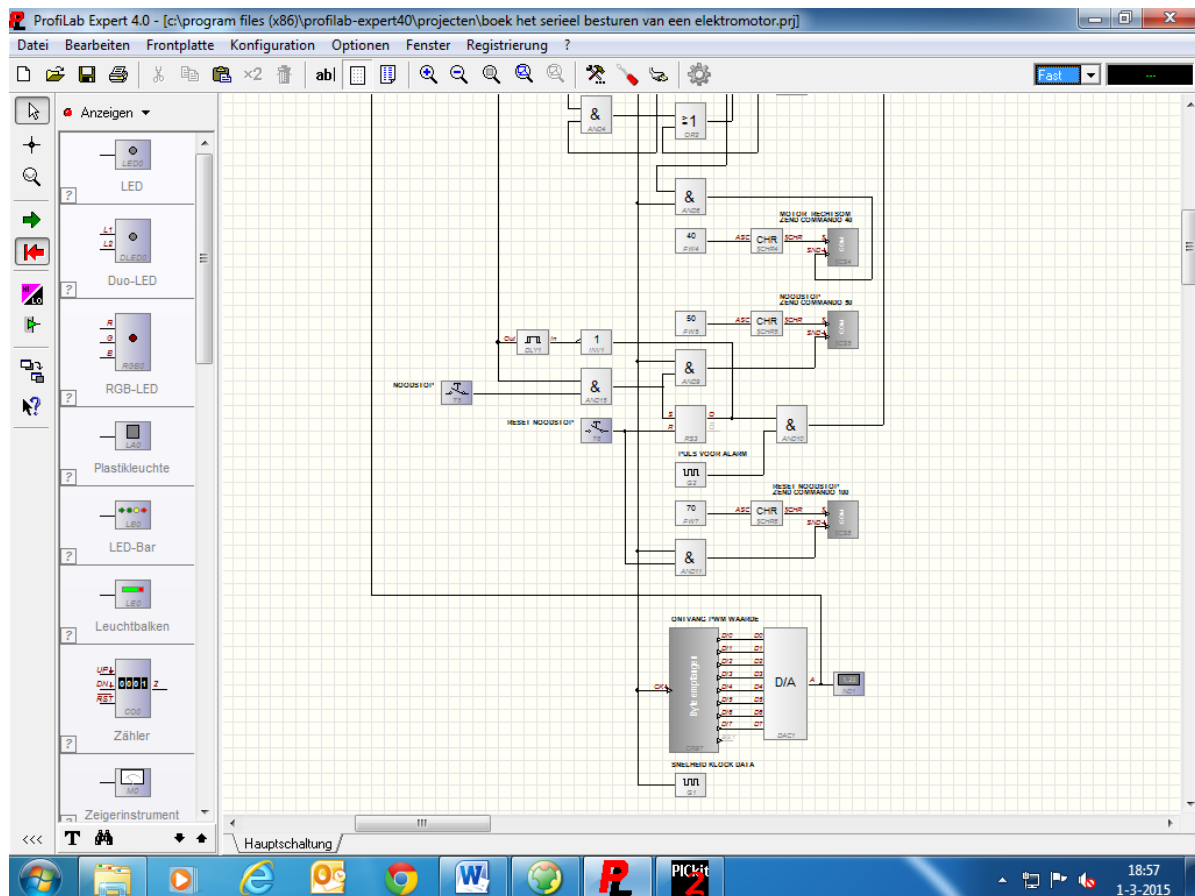
Hieronder kan je de poort zien waar het display op aangesloten kan worden.
 Het display wordt aangesloten op de tien polige connector van poort D.



We gaan nu het ProfiLab programma maken.
 Ik ga het programma in twee delen plaatsen, omdat het anders niet goed te zien is.
 Hieronder het bovenste gedeelte van het programma.



Hieronder het onderste gedeelte van het programma. Zoals je kan zien loopt het in elkaar over, dus het is redelijk te zien zo.

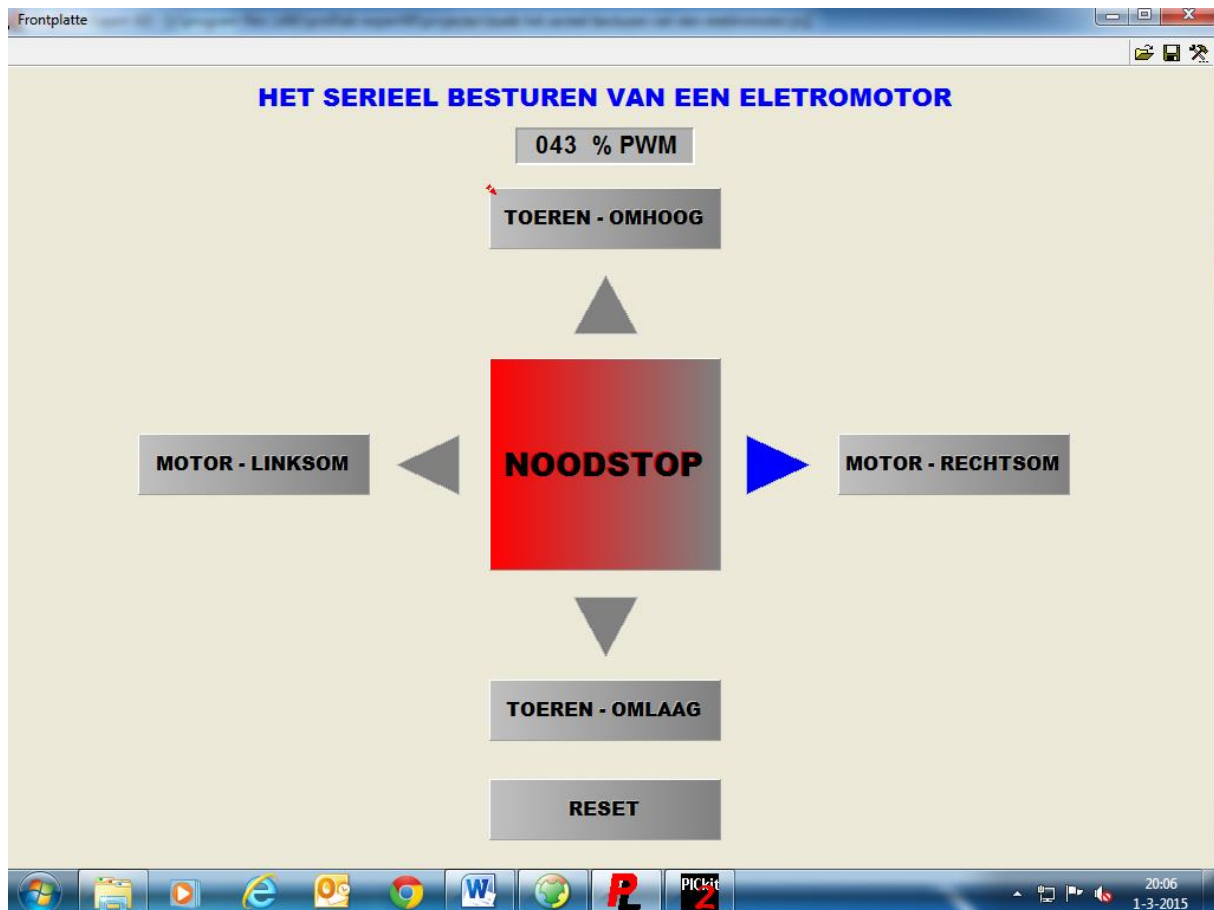


We slepen zes drukknoppen T1 tot en met T6, veertien and poorten AND1 tot en met AND14, zes or poorten OR1 tot en met OR6, vier leds LED1 tot en met LED4, zeven maal een vaste waarde module FW1 tot en met FW7, één inverter INV1, één delay module DLY1, één vergelijker module AVG1, drie maal een flipflop module RS1 tot en met RS3, zes maal decimaal naar string module \$CHR1 tot en met \$CHR6, één massa module MASSA TEKEN, zes seriële modules \$CS1 tot en met \$CS6, één maal een acht bit serieel module CRB7, één digitaal/analoog module DAC1, en één display ND1.

Ik ga bij dit voorbeeld niet vertellen hoe alles aangesloten moet worden, dat zou veel te veel tekst worden en dan wordt het er niet echt duidelijker op.

De waarde die in de modules komen zijn. In vaste waarde module FW1 komt 10, in FW2 komt 20, in FW3 komt 30, in FW4 komt 40, in FW5 komt 50, in FW6 komt 30, en in FW7 komt 70. In de delay module DLY1 komt afval vertraagd 5. De acht bit serieel module CRB7 stellen we weer in op COM5, 9600 BAUD, DATA 8, PARITY NONE, en STOP 1. De digitaal/analoog module zetten we op 0 tot en met 255, En het display ND1 zetten we op drie digit met de tekst % en pwm in het veld.

Op blz. 102 staat de frontplaat.



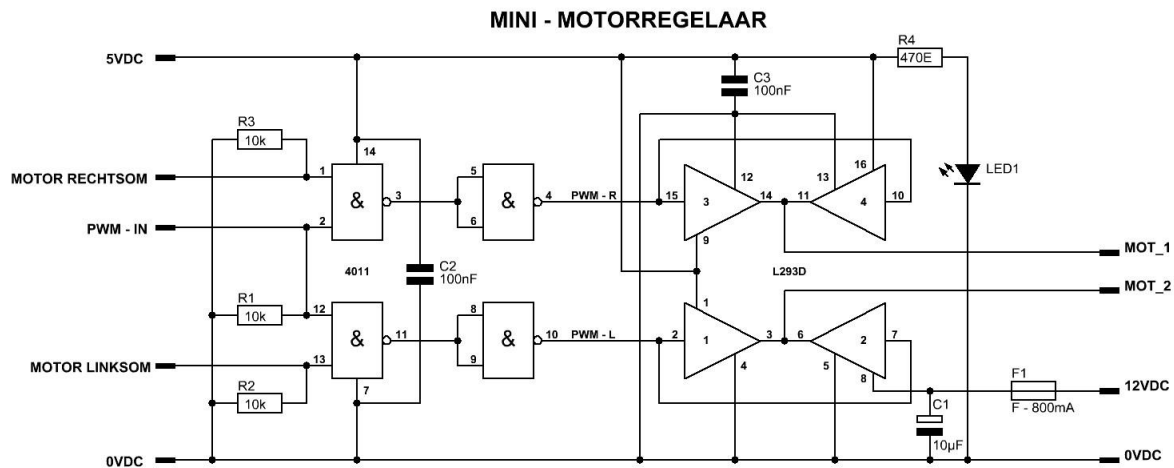
Dit is de frontplaat geworden voor de motor besturing. Met de knoppen motor linksom en motor rechtsom wordt de waarde 30 en 40 serieel verstuurd. Met de knoppen toeren omhoog en toeren omlaag wordt de waarde 10 en 20 serieel verzonden. En met de knop noodstop en reset wordt de waarde 50 en 70 serieel verzonden. Als één van deze waardes binnen komt op de controller, dan zal deze de actie uitvoeren welke bij één van de commando's hoort. Er staan ook vier led's op de frontplaat, deze lichten op als één van de vier schakelaars ingedrukt wordt, zoals je kan zien op de frontplaat is motor rechtsom in gedrukt. Bij de knoppen motor links en rechtsom blijven de led's branden, en bij toeren omhoog en toeren omlaag lichten de led's alleen op als deze knoppen ingedrukt worden. Als nu de noodstop ingedrukt wordt, dan gaan de drie led's die uit zijn knipperen, (motor links of rechtsom blijft branden) het motor toerental loopt automatisch terug naar nul. Zolang de led's knipperen kan er geen drukknop bedient worden behalve de reset knop, als deze ingedrukt wordt gaan de led's uit en worden de andere knoppen weer vrij gegeven. De knoppen motor links en rechtsom kunnen bedient worden tot een waarde van 29%, als de waarde 30% is zijn de knoppen vergrendelt. Tot die waarde kan de motor nog veilig omgepoold worden, daar boven niet meer dus vandaar de vergrendeling.

Zoals we gezien hebben gebruiken we ook twee displays, één zit er op de controller en één staat er op de frontplaat van Profilab. Het display op de frontplaat en het display aan de controller worden beide door Profilab aangestuurd. Op het display bij de controller krijgen we de zelfde informatie te zien als het display op de frontplaat. We hebben dus seriële communicatie in twee richtingen, dat is ook te zien op de seriële interface. Daar zullen bij sommige acties twee led's gaan branden.

Verder valt er niet zoveel uit te leggen over de werking en de software, ik zou zeggen gewoon nabouwen en kijk wat er gebeurt. Of bouw zelf iets naar eigen inzicht.

Om een elektromotor linksom, rechtsom, en in toeren te regelen heb je een zogenaamde H_BRUG nodig. Een H_BRUG kan je maken met bijv. een relaischakeling, maar dan is het toerental moeilijker te regelen. Er zijn ook IC's te koop waar een H_BRUG in zit bijv. de L298, de LM18298, de LMD18200, de LMD18201, enz. Ik ga een paar schema's plaatsen één met de L293D, één met losse transistoren, en één met losse mosfet's (zijn ook transistoren natuurlijk).

We beginnen met de L293D, zie schema hieronder.



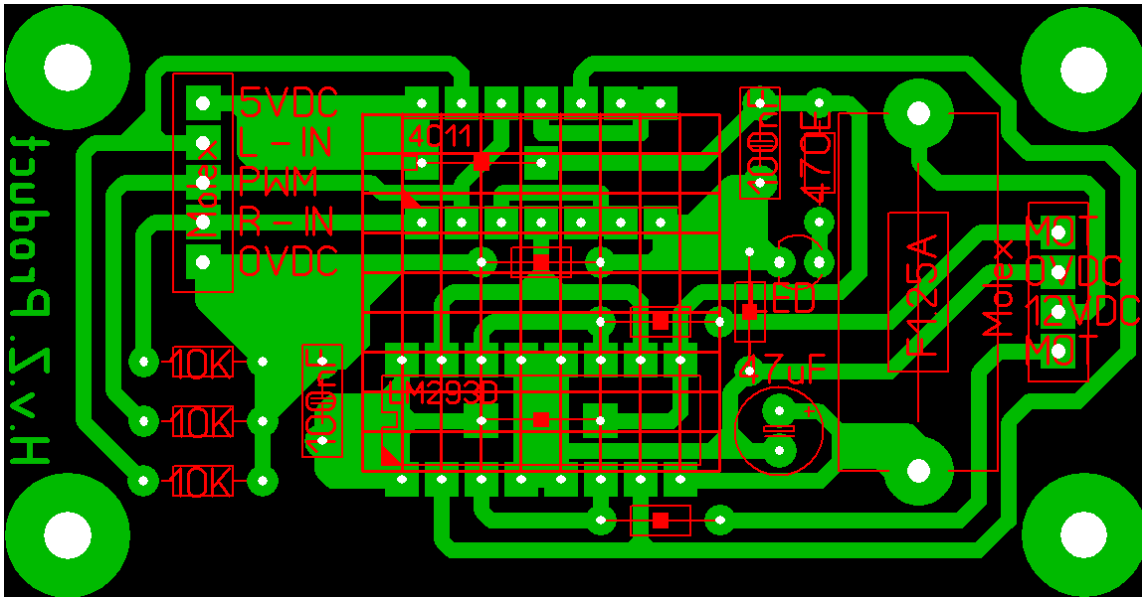
De oplettende lezer zal het schema wel herkennen dit schema is in het begin ook geplaatst, alleen wordt de regelaar nu aangesloten op het controllerboard en niet op de K8055. Ook staan de weerstanden R1, R2, en R3 anders in het schema, dat zijn nu pull down weerstanden.

Aan de linkerkant zie je de aansluitingen voor de ingangen van de regelaar. Motor rechtsom komt aan PORTC.1, motor linksom komt aan PORC.0, en pwm – in komt aan PORTC.2 van het controllerboard. Aan de rechterkant zie je de aansluitingen voor de motor en voor de voeding van de motor. Op mot_1 en mot_2 komt de motor te zitten, en op de onderste twee aansluitingen komt de voedingsspanning voor de motor. Je kan dit weer opbouwen op een bread board of op gaatjes print.

Zelf heb ik er een klein printje voor gemaakt, omdat ik deze regelaar nog wel eens gebruik in robot projecten. Ik stuur er dan bijv. een RB35 mee aan, dat is een motor met vertraging en die laat zich prima regelen door deze schakeling. Let er wel op dat de maximum pwm frequentie niet boven de 5KHz komt, want dat is de max frequentie voor de L293D. In de software voor dit voorbeeld werken we op 4KHz zoals je kan zien in het basic programma.

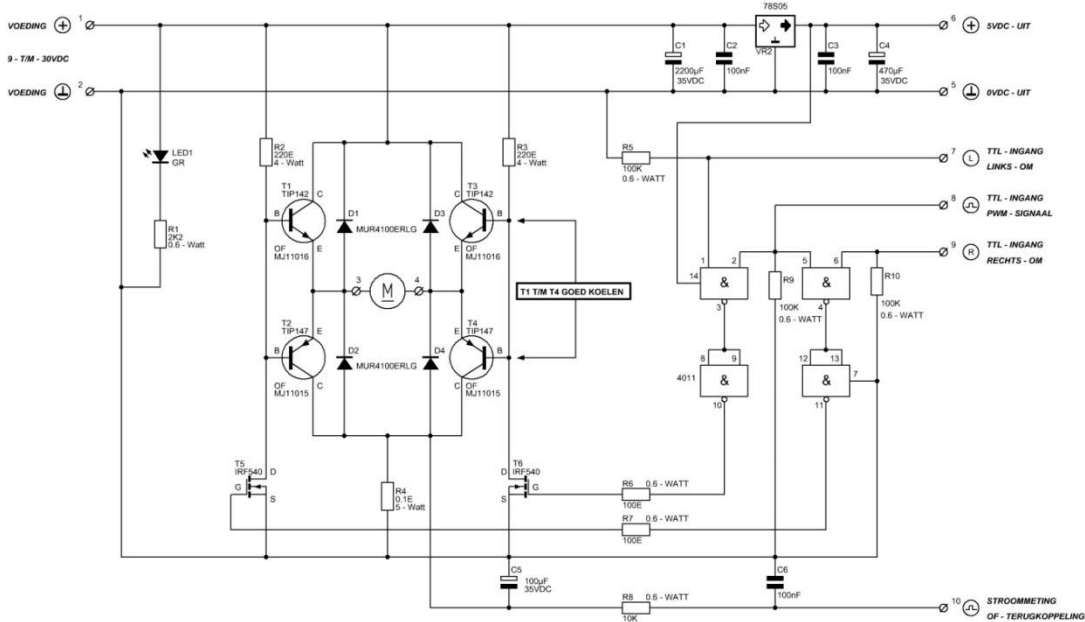
Voor meer info over de L293D kan je weer de datasheet raadplegen op het net.

Op blz. 104 staat de print voor de mini – motorregelaar.



De print heeft een maat van 73 X 38mm. De print is voorzien van Molex connectoren voor het makkelijk aansluiten op het controllerboard. Ook is er een koellichaam op de twee IC's gelijmd voor de warmte afvoer. Tot zover over de mini – motorregelaar.

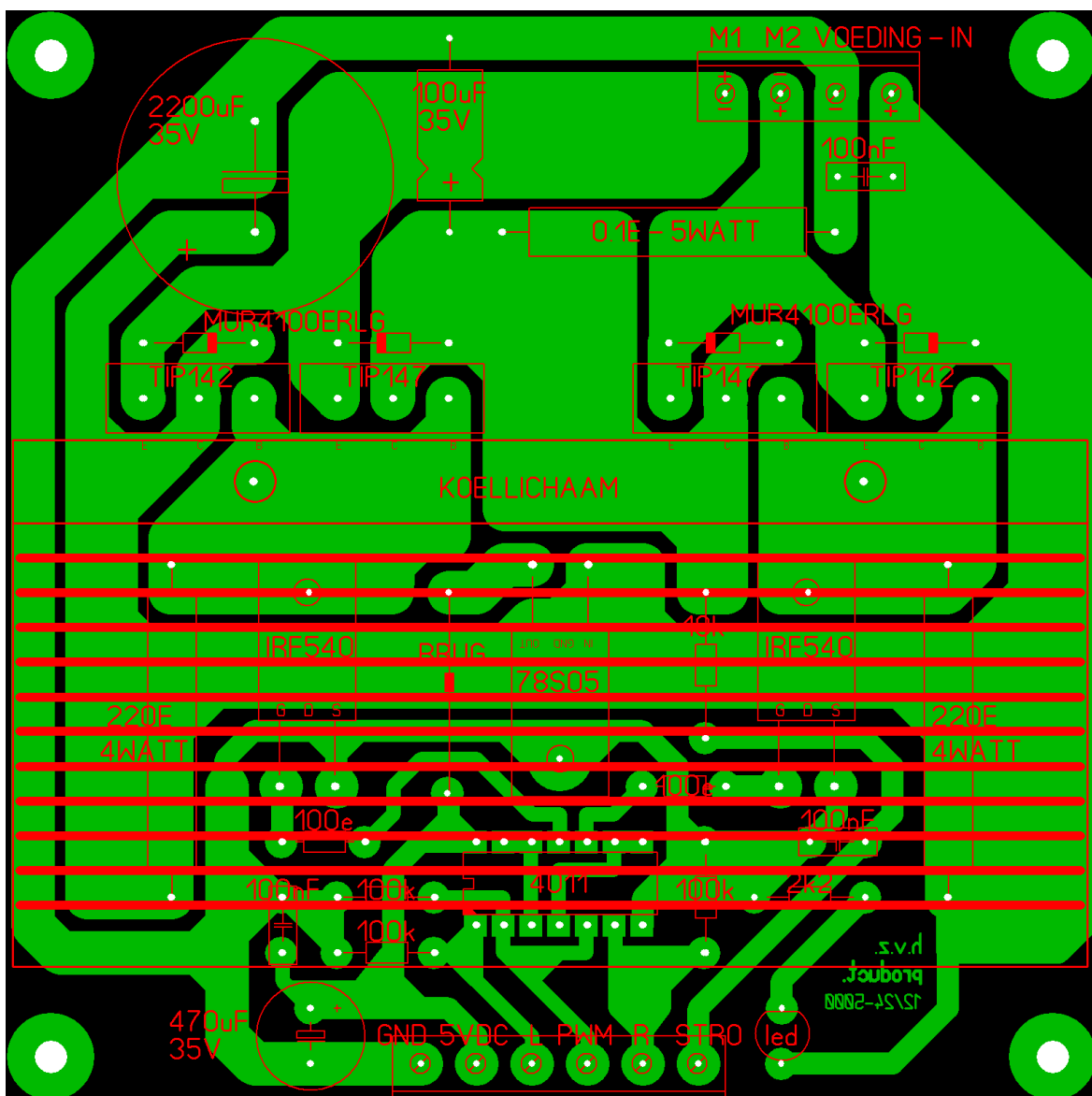
Hier onder een schema voor het regelen van een motor tot een maximum van +/- 75Watt met de TIP142 en de TIP147, en +/- 200Watt als de MJ11016 en de MJ11015 gebruikt worden .



Zoals je kan zien zijn hier darlington transistoren gebruikt voor de H – BRUG. De transistoren hangen met de emitter aan elkaar, de collector van de NPN transistor hangt aan de voedingspanning en de collector van de PNP transistor hangt aan de GND. De basissen van de PNP en de NPN transistoren hangen aan elkaar, en worden via T5 en T6 naar de GND getrokken. De twee mosfets (van het type IRF540) schakelen dus de basissen van de darlington transistoren. De gate's van de mosfets worden via de nand poort 4011 aangestuurd. Op deze poort komt het signaal van het controllerboard binnen.

Er zit ook een spanningsregelaar op de motorregelaar om eventuele externe zaken van spanning te voorzien. Ook zit er een uitgang op waarmee de stroom kan worden gemeten die door de motor gaat. Dit kan gebruikt worden als overstroom detectie, maar ook voor eventuele terug koppeling om het toerental constant te houden als de motor belast wordt. Dit signaal komt dan op de analoge ingang van de controller.

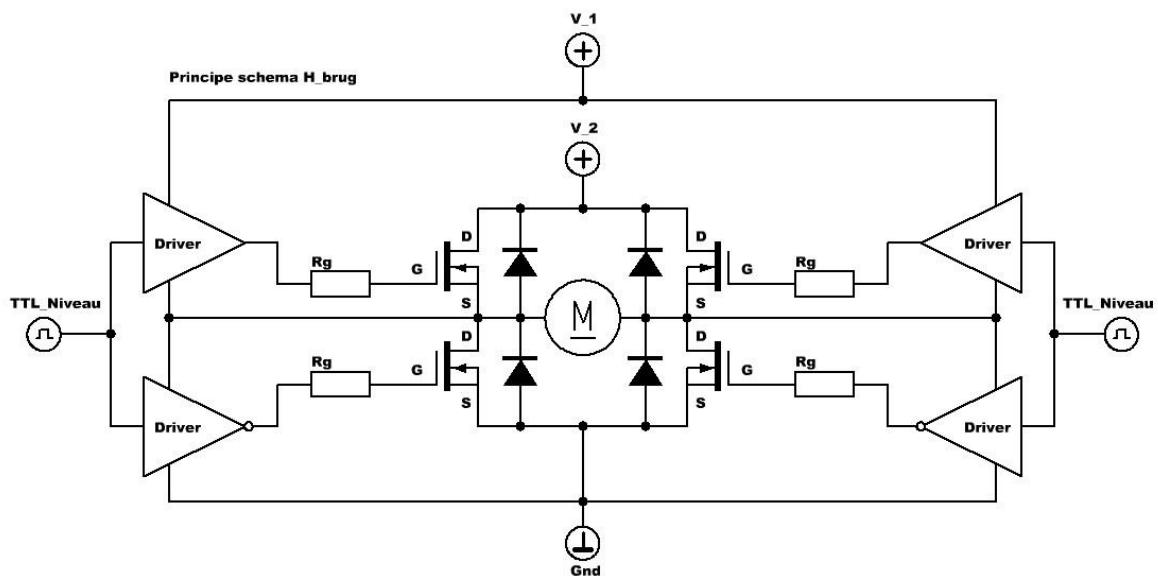
Hieronder de print van de motorregeling.



De print heeft een formaat van 100 X 100mm. Voor de regelaar met de MJ11016 en de MJ11015 kan de zelfde print gebruikt worden, alleen zit er dan een ander koellichaam op. Dat koellichaam moet dan wel aangepast worden, omdat de MJ11016 en de MJ11015 in een TO3 behuizing zitten. Het koellichaam dat gebruikt wordt is 100 X 100mm, en daar worden dan stukken van de koelvinnen weg gevreesd om de transistoren pas te maken. Tot zover over de 75W / 200W regelaar.

We gaan nu een regelaar bekijken waarbij de H – BRUG is opgebouwd met mosfets, met dit principe heb je minder verliezen, dus verstoek je ook minder warmte in de regelaar.

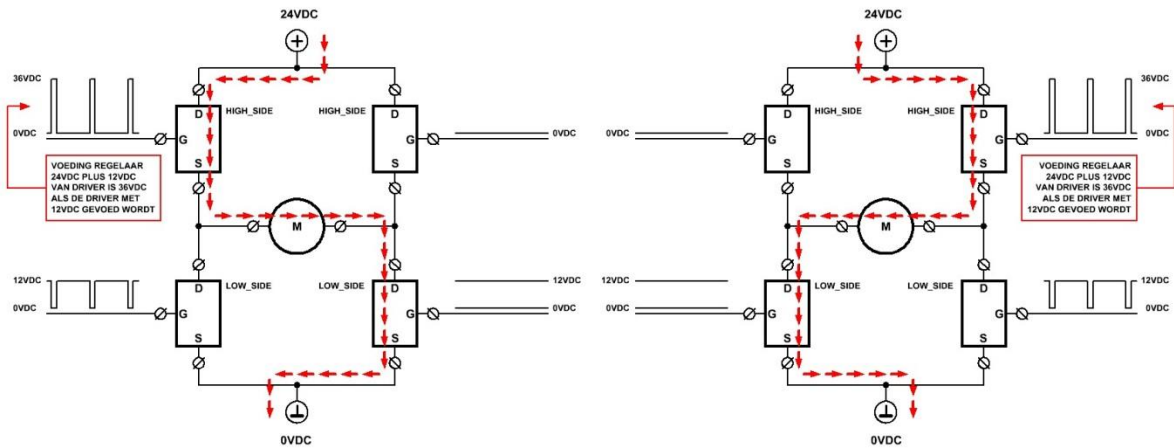
Hier onder het principe schema voor een H – BRUG.



De opbouw van de H - BRUG gebeurt met vier mosfets. De bovenste twee mosfets worden high side mosfets genoemd, en de onderste twee low side mosfets.

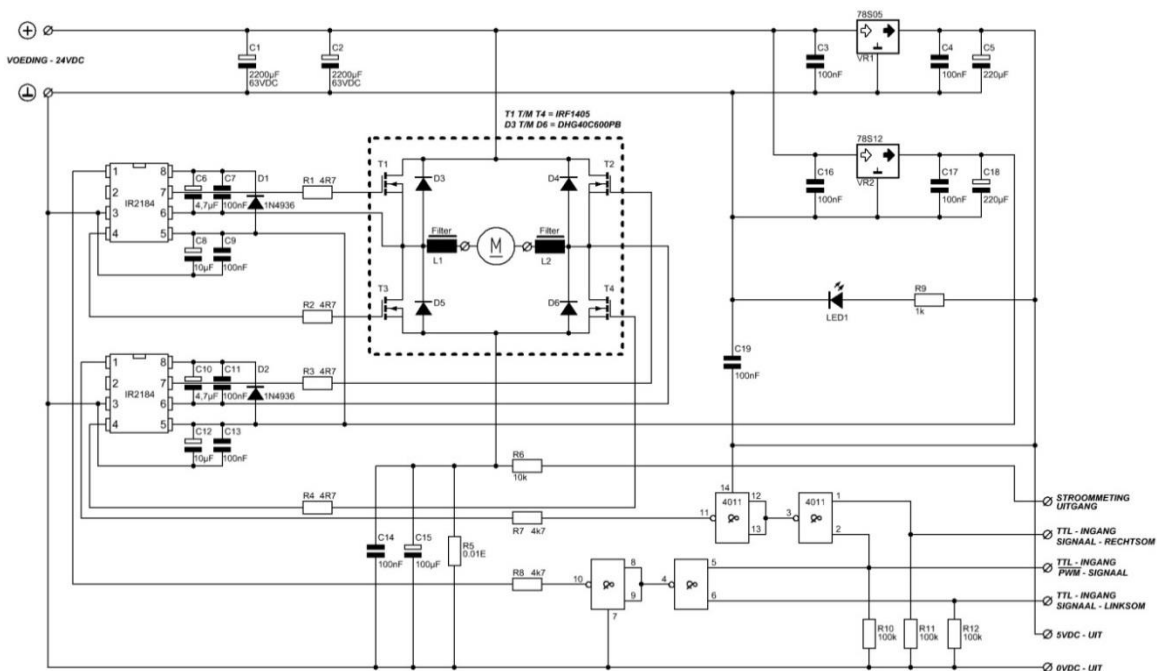
De brug kan op verschillende manieren opgebouwd worden. Het kan met p mosfets aan de bovenkant, en n mosfets aan de onderkant. Maar het kan ook met vier n mosfets, zoals in het principe schema getoond wordt. Daar worden ook speciale drivers gebruikt, deze zorgen ervoor dat de aansturing van de fets netjes verloopt. Sommige drivers bezitten een zogenaamde bootstrap, deze trap zorgt ervoor dat de high side mosfet een stukje opgebeurd wordt aan de gate. Dit is nodig, omdat de fet anders niet in geleiding komt.

Op blz. 107 een voorbeeld van het signaal dat op de gate komt. De spanning die hier getoond wordt (in dit geval 36VDC) hangt wel af van de voedingsspanning van de driver, de driver wordt in dit geval gevoed met 12VDC. De voedingsspanning van de motor wordt hier bij de voedingsspanning van de driver opgeteld, dan krijg je dus een 36V gate spanning op de bovenste fets.

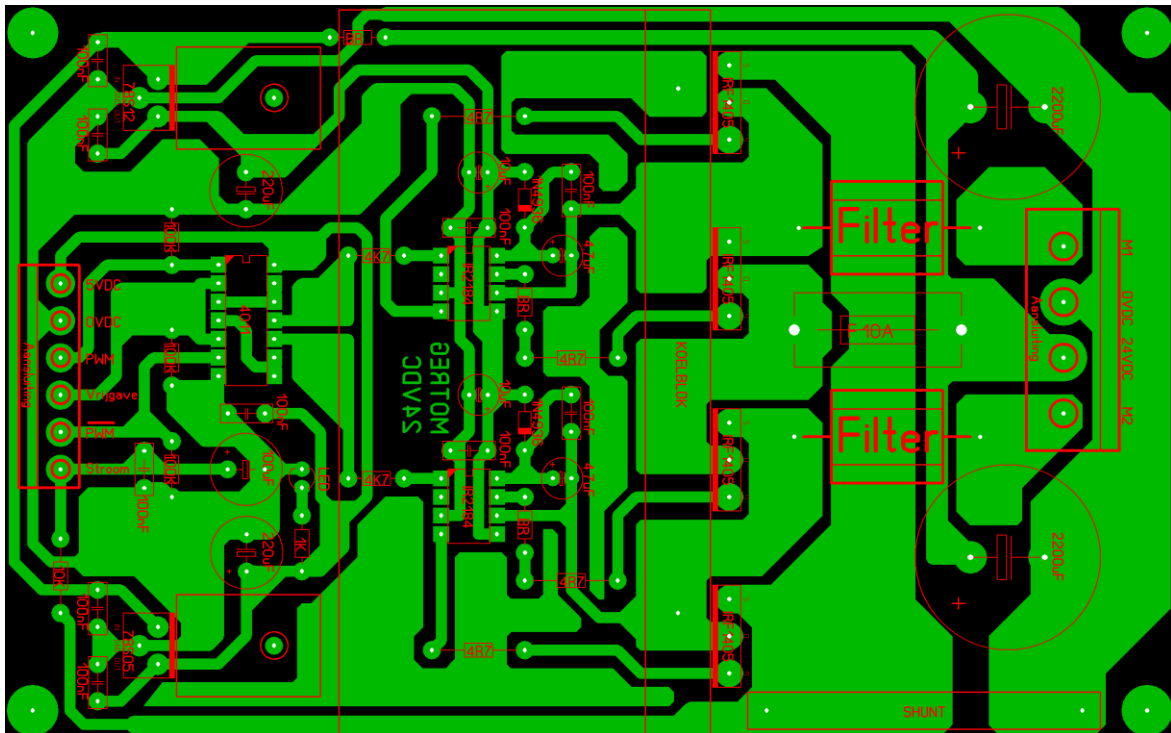


Op de afbeelding (boven) is ook het stroomverloop te zien. Als men op het linker voorbeeld kijkt, is te zien dat de high side fet (linksboven) en de low side fet (rechtsonder) in geleiding is. Je ziet dat de stroom via de bovenste fet door de motor en via de onderste fet naar de gnd loopt. Op het plaatje zijn ook de bijbehorende signalen te zien; de motor draait bijv. nu rechtsom. Op het voorbeeld rechts is de high side fet (rechtsboven) en de low side fet (linksonder) in geleiding. Je ziet dat de stroom richting nu de andere kant op loopt, de motor draait nu linksom. Als er op de ingang van de driver een pwm signaal gezet wordt, kan ook het toerental van de motor geregeld worden. De drivers kunnen direct vanuit het controllerboard aangestuurd worden, op ttl niveau dus. Tussen de driver en de fets staan weerstanden getekend. Dit worden de gate weerstanden genoemd. De weerstand waarde kan variëren van 0 tot 100 Ohm, maar dat ligt geheel aan het ontwerp van de schakeling.

Hieronder het schema van de H – BRUG met mosfets.



Over de fets staan diodes om de fets te beschermen; dit vanwege de inductieve belasting die een elektromotor is. In de meeste fets zijn de diodes wel aanwezig, maar het heeft echt de voorkeur deze externe diodes erbij te plaatsen. Het moeten wel snelle diodes zijn, dus geen diodes die gebruikt worden voor brugcellen (gelijkrichters.) Verder kan er nog met filters gewerkt worden, maar dat is niet speciaal nodig. Ook moeten de fets van een koellichaam worden voorzien, om te zorgen dat deze niet te warm worden. Er kan ook nog een ptc of ntc op de koelplaat gemonteerd worden om de temperatuur te controleren.



Ik heb ook een print gemaakt voor deze motorregeling. Het formaat voor deze print is 100 X 160mm.

Er zijn twee filters opgenomen om eventuele storingen tegen te gaan. De filters zijn opgebouwd rond twee ferriet kernen. Er is één maal een draad doorgehaald met een diameter van 2.5mm².

Ook is er een zekering opgenomen op de print als extra bescherming tegen overstroom. Ook wordt de stroom detectie gebruikt om de stroom binnen de perken te houden.

Tot zover dit onderwerp over het serieel besturen van een elektromotor.

6.8: Relaischakeling met XBEE

We gaan nu een relaischakeling maken met Xbee modules. We beginnen weer met het basic programma, en daarna het Profilab programma. Later gaan we het over de Xbee modules hebben.

```
'* Name      : RELAISSCHAKELING MET XBEE.
'* Author    : H van Zwieten.
'* Notice    : Copyright (c) 2015 H.v.Z.
'*          : All Rights Reserved
'* Date      : 3-3-2015
'* Version   : 1.0
'* Notes     : SOFTWARE VOOR HET BOEK.

;-----
; DE XBEE ROUTER (ONTVANGER) MOET OP API MODE STAAN.
; DATA D0, D1, D2, D3, D4, D6, D7, D11, D12 MOET OP 4 STAAN,
; DAN ZIJN DE UITGANGEN IN RUST LAAG.
;-----

;-----
; DE XBEE COORDINATOR (ZENDER) MOET OP API MODE STAAN.
; HET PAN_ID MOET ZOWEL BIJ DE ROUTER ALS DE COORDINATOR HET ZELFDE ZIJN.
; DE LED BIJ DE COORDINATOR KNIPPERT LANGZAAM, EN BIJ DE ROUTER SNEL.
; D_11 = P1 IN XCTU, EN D_12 = P2 IN XCTU.
; D_11 = D_11 OP DE X_BEE UITGANGS PRINT.
; D_0  = D_0  OP DE X_BEE UITGANGS PRINT.
; D_1  = D_1  OP DE X_BEE UITGANGS PRINT.
; D_2  = D_2  OP DE X_BEE UITGANGS PRINT.
; D_3  = D_3  OP DE X_BEE UITGANGS PRINT.
; D_6  = D_6  OP DE X_BEE UITGANGS PRINT.
; D_7  = D_7  OP DE X_BEE UITGANGS PRINT.
; D_4  = D_4  OP DE X_BEE UITGANGS PRINT.
;-----

Device 16F887                ; Processor type

Xtal 10                    ; Cristal 10Mhz

Asm
CONFIG_REQ
__CONFIG __CONFIG1, HS_OSC & WDT_OFF & DEBUG_OFF & FCMEN_OFF & LVP_OFF &
IESO_OFF & BOR_OFF & CPD_OFF & CP_OFF & MCLRE_OFF & PWRTE_ON
__CONFIG __CONFIG2, WRT_OFF & BOR40V
EndAsm

All_Digital true           ; Alle poorten digitaal

Declare LCD_RSPin PORTD.2    ; Reset display poort D.2
Declare LCD_ENPin PORTD.3    ; Enable display poort D.3
Declare LCD_DTPin PORTD.4    ; Data display poort D.4 t/m D.7

Declare Hserial_Baud 9600    ; Set baudrate op 9600

Declare Hserial_RCSTA %10010000 ; Register instelling
Declare Hserial_TXSTA %10100110 ; Register instelling
```

```

Declare Hserial_Clear = On           ; Reset buffer bij een overflow

Declare Serial_Baud 9600           ; Baudrate 9600

Declare Rsin_Pin PORTB.0           ; Data in
Declare Rsout_Pin PORTB.1         ; Data uit

Symbol TIJD = 100                 ; Waarde pauze

Dim DATA_IN As Byte           ; Data serieelepoort

Cls                               ; Wis display

DelayMS 500                       ; Pauze 0.5 sec

Clear                              ; Wis geheugen

    ;543210                         ; Hulpregel poort A
PORTA = %000000                   ; Maak poort A laag
TRISA = %111111                   ; Poort_A I/O

    ;543210                         ; Hulpregel poort B
PORTB = %000000                   ; Maak poort B laag
TRISB = %000001                   ; Poort_B I/O

    ;76543210                       ; Hulpregel poort C
PORTC = %00000000                 ; Maak poort C laag
TRISC = %00000000                 ; Poort_C I/O

    ;76543210                       ; Hulpregel poort D
PORTD = %00000000                 ; Maak poort D laag
TRISD = %00000000                 ; Poort_D I/O

    ;210                             ; Hulpregel poort E
PORTE = %000                     ; Maak poort E laag
TRISE = %111                     ; Poort_E I/O

    ;76543210                       ; Hulpregel analoog
ADCON0 = %00000001                ; ADCON0 register analoog

    ;543210                         ; Hulpregel analoog
ANSELH = %000000                 ; Ansel register analoog

;-----
; CODE VOOR XBEE UITGANGEN - ZONDER EXPLECIT ADRESS.
;-----

; DIGITALE UITGANG D_0 - AAN
; 0X7E,0X00,0X10,0X17,0X01,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0XFF,0XFF,0XFF,
; 0XFE,0X02,0X44,0X30,0X05,0X71
; DIGITALE UITGANG D_0 - UIT
; 0X7E,0X00,0X10,0X17,0X01,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0XFF,0XFF,0XFF,
; 0XFE,0X02,0X44,0X30,0X04,0X72

```

```

; DIGITALE UITGANG D_1 - AAN
; 0X7E,0X00,0X10,0X17,0X01,0X00,0X00,0X00,0X00,0X00,0X00,0XFF,0XFF,0XFF,
; 0XFE,0X02,0X44,0X31,0X05,0X70
; DIGITALE UITGANG D_1 - UIT
; 0X7E,0X00,0X10,0X17,0X01,0X00,0X00,0X00,0X00,0X00,0X00,0XFF,0XFF,0XFF,
; 0XFE,0X02,0X44,0X31,0X04,0X71

; DIGITALE UITGANG D_2 - AAN
; 0X7E,0X00,0X10,0X17,0X01,0X00,0X00,0X00,0X00,0X00,0X00,0XFF,0XFF,0XFF,
; 0XFE,0X02,0X44,0X32,0X05,0X6F
; DIGITALE UITGANG D_2 - UIT
; 0X7E,0X00,0X10,0X17,0X01,0X00,0X00,0X00,0X00,0X00,0X00,0XFF,0XFF,0XFF,
; 0XFE,0X02,0X44,0X32,0X04,0X70

; DIGITALE UITGANG D_3 - AAN
; 0X7E,0X00,0X10,0X17,0X01,0X00,0X00,0X00,0X00,0X00,0X00,0XFF,0XFF,0XFF,
; 0XFE,0X02,0X44,0X33,0X05,0X6E
; DIGITALE UITGANG D_3 - UIT
; 0X7E,0X00,0X10,0X17,0X01,0X00,0X00,0X00,0X00,0X00,0X00,0XFF,0XFF,0XFF,
; 0XFE,0X02,0X44,0X33,0X04,0X6F

; DIGITALE UITGANG D_4 - AAN
; 0X7E,0X00,0X10,0X17,0X01,0X00,0X00,0X00,0X00,0X00,0X00,0XFF,0XFF,0XFF,
; 0XFE,0X02,0X44,0X34,0X05,0X6D
; DIGITALE UITGANG D_4 - UIT
; 0X7E,0X00,0X10,0X17,0X01,0X00,0X00,0X00,0X00,0X00,0X00,0XFF,0XFF,0XFF,
; 0XFE,0X02,0X44,0X34,0X04,0X6E

; DIGITALE UITGANG D_6 - AAN
; 0X7E,0X00,0X10,0X17,0X01,0X00,0X00,0X00,0X00,0X00,0X00,0XFF,0XFF,0XFF,
; 0XFE,0X02,0X44,0X36,0X05,0X6B
; DIGITALE UITGANG D_6 - UIT
; 0X7E,0X00,0X10,0X17,0X01,0X00,0X00,0X00,0X00,0X00,0X00,0XFF,0XFF,0XFF,
; 0XFE,0X02,0X44,0X36,0X04,0X6C

; DIGITALE UITGANG D_7 - AAN
; 0X7E,0X00,0X10,0X17,0X01,0X00,0X00,0X00,0X00,0X00,0X00,0XFF,0XFF,0XFF,
; 0XFE,0X02,0X44,0X37,0X05,0X6A
; DIGITALE UITGANG D_7 - UIT
; 0X7E,0X00,0X10,0X17,0X01,0X00,0X00,0X00,0X00,0X00,0X00,0XFF,0XFF,0XFF,
; 0XFE,0X02,0X44,0X37,0X04,0X6B

; DIGITALE UITGANG D_11 - AAN
; 0X7E,0X00,0X10,0X17,0X01,0X00,0X00,0X00,0X00,0X00,0X00,0XFF,0XFF,0XFF,
; 0XFE,0X02,0X50,0X31,0X05,0X64
; DIGITALE UITGANG D_11 - UIT
; 0X7E,0X00,0X10,0X17,0X01,0X00,0X00,0X00,0X00,0X00,0X00,0XFF,0XFF,0XFF,
; 0XFE,0X02,0X50,0X31,0X04,0X65

;-----
; CODE VOOR XBEE UITGANGEN - MET EXPLICIT ADRESS - ADRES ONTVANGER -
; 13A2004092C0F2
;-----

; DIGITALE UITGANG D_0 - AAN
; 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,0XF2,0XFF,
; 0XFE,0X02,0X44,0X30,0X05,0X36

```

```

; DIGITALE UITGANG D_0 - UIT
; 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,0XF2,0XFF,
; 0XFE,0X02,0X44,0X30,0X04,0X37

; DIGITALE UITGANG D_1 - AAN
; 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,0XF2,0XFF,
; 0XFE,0X02,0X44,0X31,0X05,0X35
; DIGITALE UITGANG D_1 - UIT
; 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,0XF2,0XFF,
; 0XFE,0X02,0X44,0X31,0X04,0X36

; DIGITALE UITGANG D_2 - AAN
; 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,0XF2,0XFF,
; 0XFE,0X02,0X44,0X32,0X05,0X34
; DIGITALE UITGANG D_2 - UIT
; 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,0XF2,0XFF,
; 0XFE,0X02,0X44,0X32,0X04,0X35

; DIGITALE UITGANG D_3 - AAN
; 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,0XF2,0XFF,
; 0XFE,0X02,0X44,0X33,0X05,0X33
; DIGITALE UITGANG D_3 - UIT
; 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,0XF2,0XFF,
; 0XFE,0X02,0X44,0X33,0X04,0X34

; DIGITALE UITGANG D_4 - AAN
; 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,0XF2,0XFF,
; 0XFE,0X02,0X44,0X34,0X05,0X32
; DIGITALE UITGANG D_4 - UIT
; 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,0XF2,0XFF,
; 0XFE,0X02,0X44,0X34,0X04,0X33

; DIGITALE UITGANG D_6 - AAN
; 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,0XF2,0XFF,
; 0XFE,0X02,0X44,0X36,0X05,0X30
; DIGITALE UITGANG D_6 - UIT
; 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,0XF2,0XFF,
; 0XFE,0X02,0X44,0X36,0X04,0X31

; DIGITALE UITGANG D_7 - AAN
; 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,0XF2,0XFF,
; 0XFE,0X02,0X44,0X37,0X05,0X2F
; DIGITALE UITGANG D_7 - UIT
; 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,0XF2,0XFF,
; 0XFE,0X02,0X44,0X37,0X04,0X30

; DIGITALE UITGANG D_11 - AAN
; 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,0XF2,0XFF,
; 0XFE,0X02,0X50,0X31,0X05,0X29
; DIGITALE UITGANG D_11 - UIT
; 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,0XF2,0XFF,
; 0XFE,0X02,0X50,0X31,0X04,0X2A

;-----
; PROGRAMMA CODE VOOR RELAISSCHAKELING MET XBEE - MET EXPLECIT ADRESS -
; 13A2004092C0F2
;-----

```



```

RUN:
    DATA_IN = RSIn

; DIGITALE UITGANG D_11 - AAN
If DATA_IN = 1 Then
    HRSOut 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,
        0XF2,0XFF,0XFE,0X02,0X50,0X31,0X05,0X29
    Print At 1,1,"D0=1"
    DelayMS TIJD
EndIf

; DIGITALE UITGANG D_11 - UIT
If DATA_IN = 10 Then
    HRSOut 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,
        0XF2,0XFF,0XFE,0X02,0X50,0X31,0X04,0X2A
    Print At 1,1,"D0=0"
    DelayMS TIJD
EndIf

;DIGITALE UITGANG D_0 - AAN
If DATA_IN = 2 Then
    HRSOut 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,
        0XF2,0XFF,0XFE,0X02,0X44,0X30,0X05,0X36
    Print At 1,5,"D1=1"
    DelayMS TIJD
EndIf

; DIGITALE UITGANG D_0 - UIT
If DATA_IN = 20 Then
    HRSOut 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,
        0XF2,0XFF,0XFE,0X02,0X44,0X30,0X04,0X37
    Print At 1,5,"D1=0"
    DelayMS TIJD
EndIf

; DIGITALE UITGANG D_1 - AAN
If DATA_IN = 3 Then
    HRSOut 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,
        0XF2,0XFF,0XFE,0X02,0X44,0X31,0X05,0X35
    Print At 1,9,"D2=1"
    DelayMS TIJD
EndIf

; DIGITALE UITGANG D_1 - UIT
If DATA_IN = 30 Then
    HRSOut 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,
        0XF2,0XFF,0XFE,0X02,0X44,0X31,0X04,0X36
    Print At 1,9,"D2=0"
    DelayMS TIJD
EndIf

; DIGITALE UITGANG D_2 - AAN
If DATA_IN = 4 Then
    HRSOut 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,
        0XF2,0XFF,0XFE,0X02,0X44,0X32,0X05,0X34
    Print At 1,13,"D3=1"
    DelayMS TIJD
EndIf

```

```

; DIGITALE UITGANG D_2 - UIT
If DATA_IN = 40 Then
    HRSOut 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,
        0XF2,0XFF,0XFE,0X02,0X44,0X32,0X04,0X35
    Print At 1,13,"D3=0"
    DelayMS TIJD
EndIf

; DIGITALE UITGANG D_3 - AAN
If DATA_IN = 5 Then
    HRSOut 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,
        0XF2,0XFF,0XFE,0X02,0X44,0X33,0X05,0X33
    Print At 2,1,"D4=1"
    DelayMS TIJD
EndIf

; DIGITALE UITGANG D_3 - UIT
If DATA_IN = 50 Then
    HRSOut 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,
        0XF2,0XFF,0XFE,0X02,0X44,0X33,0X04,0X34
    Print At 2,1,"D4=0"
    DelayMS TIJD
EndIf

;DIGITALE UITGANG D_6 - AAN
If DATA_IN = 6 Then
    HRSOut 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,
        0XF2,0XFF,0XFE,0X02,0X44,0X36,0X05,0X30
    Print At 2,5,"D5=1"
    DelayMS TIJD
EndIf

; DIGITALE UITGANG D_6 - UIT
If DATA_IN = 60 Then
    HRSOut 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,
        0XF2,0XFF,0XFE,0X02,0X44,0X36,0X04,0X31
    Print At 2,5,"D5=0"
    DelayMS TIJD
EndIf

; DIGITALE UITGANG D_7 - AAN
If DATA_IN = 7 Then
    HRSOut 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,
        0XF2,0XFF,0XFE,0X02,0X44,0X37,0X05,0X2F
    Print At 2,9,"D6=1"
    DelayMS TIJD
EndIf

; DIGITALE UITGANG D_7 - UIT
If DATA_IN = 70 Then
    HRSOut 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,
        0XF2,0XFF,0XFE,0X02,0X44,0X37,0X04,0X30
    Print At 2,9,"D6=0"
    DelayMS TIJD
EndIf

```

```

; DIGITALE UITGANG D_4 - AAN
If DATA_IN = 8 Then
  HRSOut 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,
    0XF2,0XFF,0XFE,0X02,0X44,0X34,0X05,0X32
  Print At 2,13,"D7=1"
  DelayMS TIJD
EndIf

; DIGITALE UITGANG D_4 - UIT
If DATA_IN = 80 Then
  HRSOut 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,
    0XF2,0XFF,0XFE,0X02,0X44,0X34,0X04,0X33
  Print At 2,13,"D7=0"
  DelayMS TIJD
EndIf

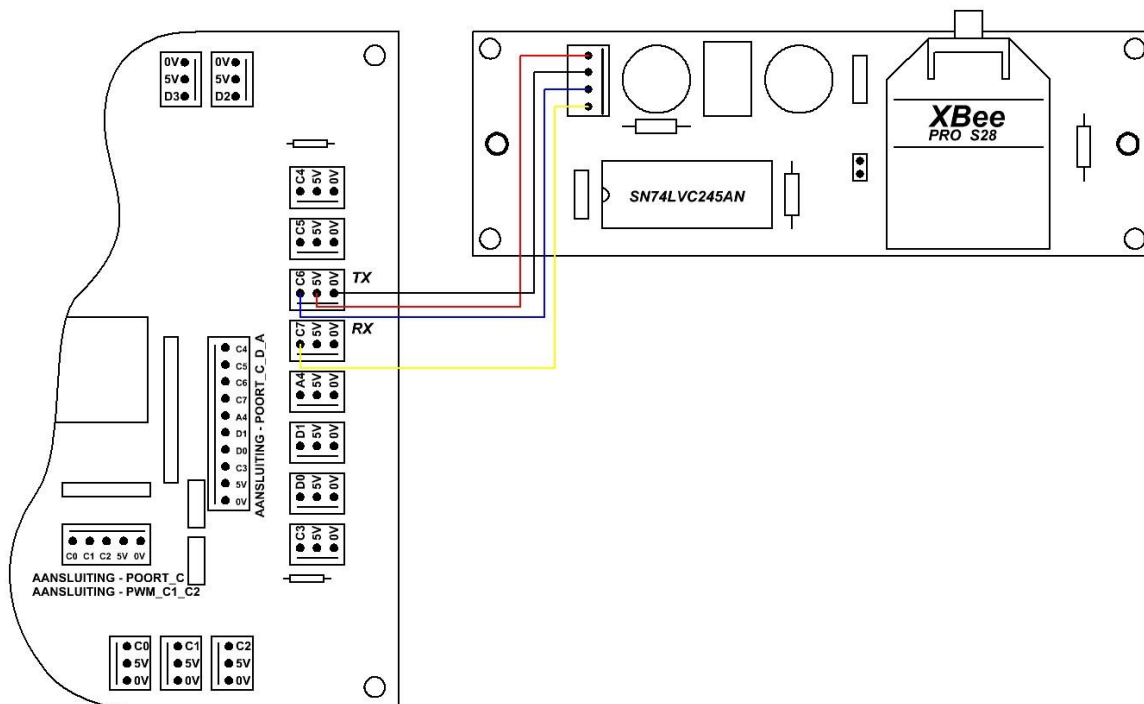
GoTo RUN

End

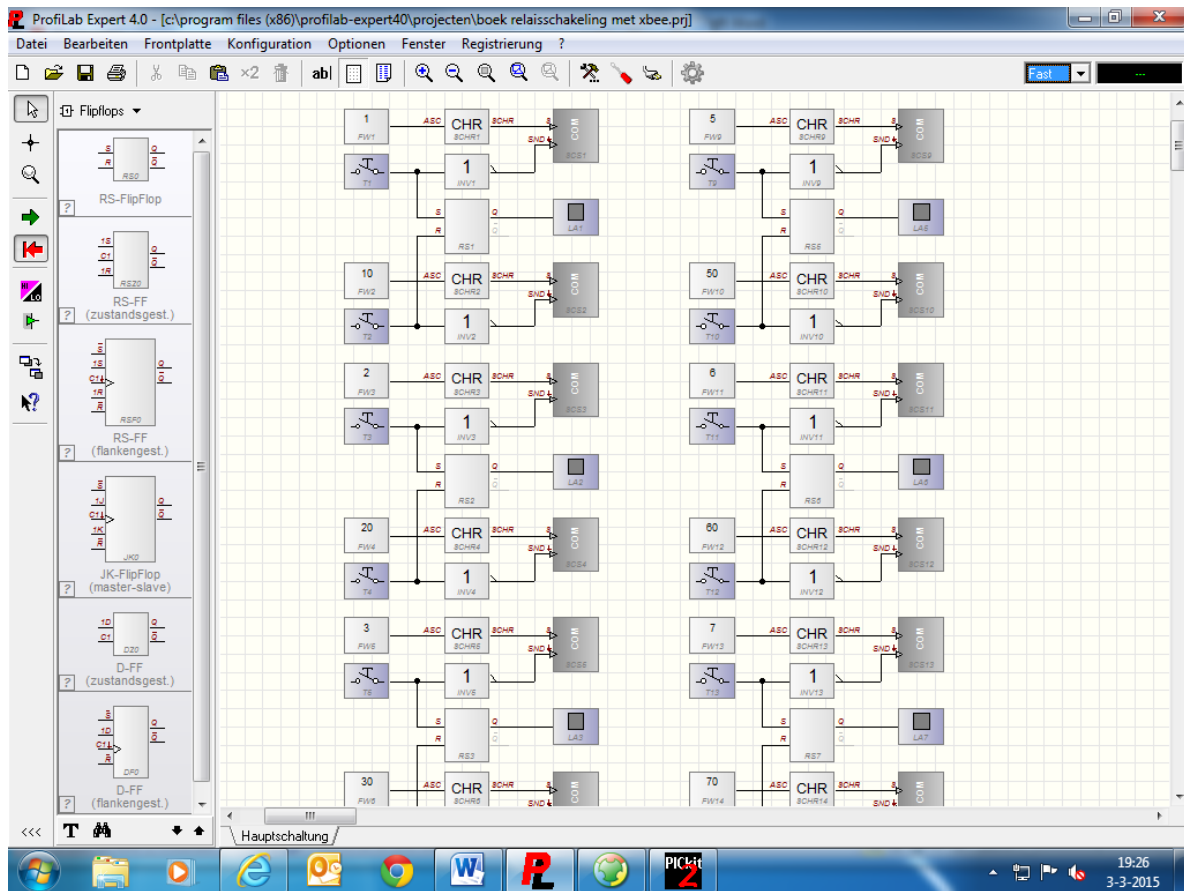
```

Zoals je kan zien in de code wordt er nu met twee seriële signalen gewerkt. Een signaal is voor de data overdracht van Profilab naar het controllerboard, en één is van het controllerboard naar de Xbee zend module. Hier onder het aansluitschema.

Zoals je op het schema kan zien staat PORTC.6 op TX, en PORTC.7 op RX. Later gaan we dieper in op de Xbee module die we gebruiken.



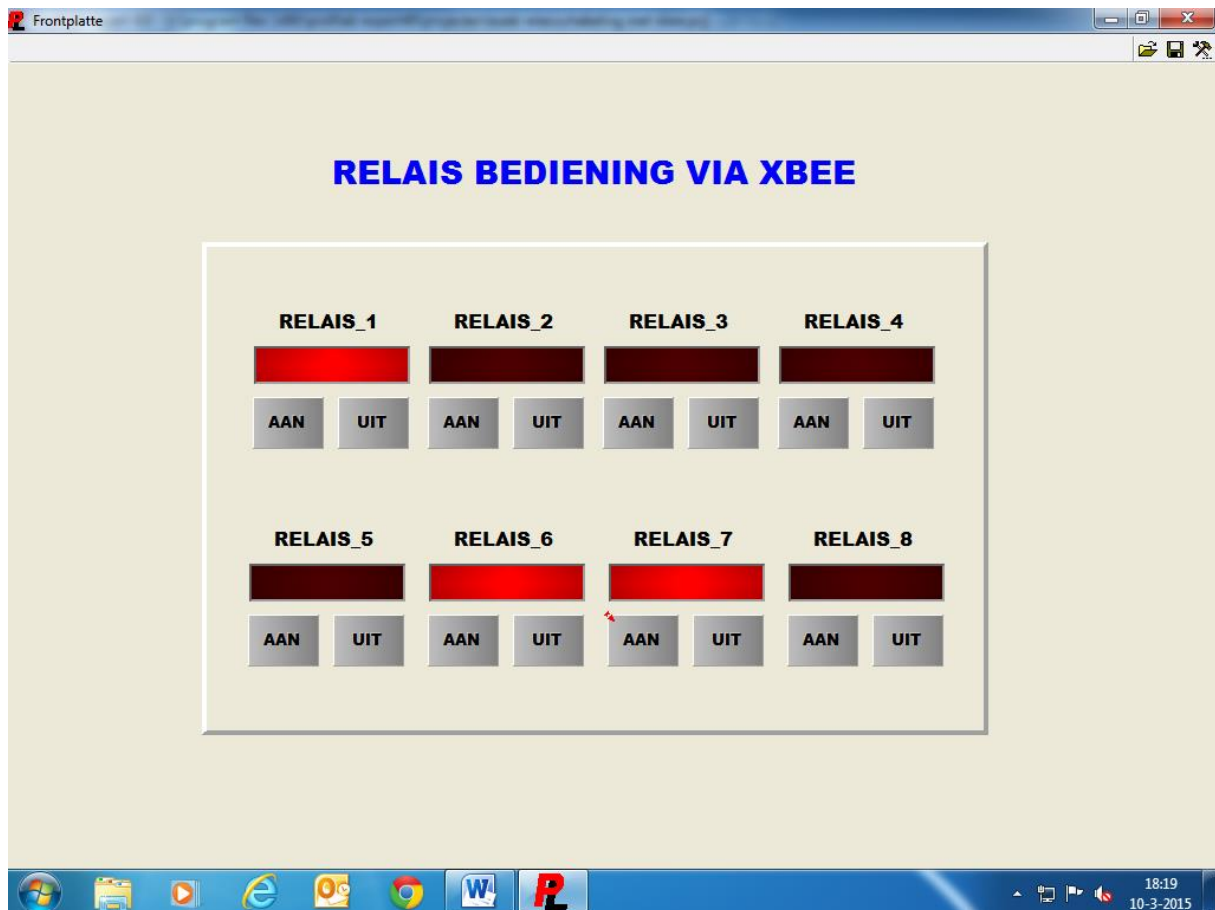
Hieronder het ProfiLab programma.



We slepen zestien maal een drukknop, zestien maal een vaste waarde module, zestien maal een ascii naar string module, zestien maal een inverter, zestien maal een serieel module, acht maal een flipflop module, en acht maal een lamp naar het werkblad. We gaan weer één gedeelte maken, en acht maal kopiëren. Drukknop T1 komt aan de ingang van inverter INV1 en aan de S ingang van de flipflop module RS1. Drukknop T2 komt aan de ingang van inverter INV2 en aan de R ingang van de flipflop module RS1. De lamp LA1 komt aan de uitgang Q van de flipflop module RS1. Vaste waarde module FW1 komt via de ascii naar string module \$CHR1 aan de \$ ingang van de seriële module \$CS1. Vaste waarde module FW2 komt via de ascii naar string module \$CHR2 aan de \$ ingang van de seriële module \$CS2. De uitgangen van de inverter INV1 en INV2 komen aan de SND ingangen van de seriële modules \$CS1 en \$CS2.

Dit gedeelte kan weer acht maal gekopieerd worden en daarna geplakt. Als je nu naar de frontplaat gaat dan staan alle drukknoppen en lampen weer boven op elkaar, het lijkt er dus op dat er van elk maar één op de frontplaat staat. Dit kan je gewoon weer verslepen en netjes op de frontplaat zetten.

We gaan nu de frontplaat weer indelen, zie blz. 117.



Zo heb ik de frontplaat ingedeeld, de drukknoppen zijn weer van tekst voorzien. Als er nu op de eerste drukknop AAN gedrukt wordt, wordt de code één naar het controllerboard verzonden. De code wordt daar omgezet (door het basic programma) naar.... `DATA_IN = RSIn`

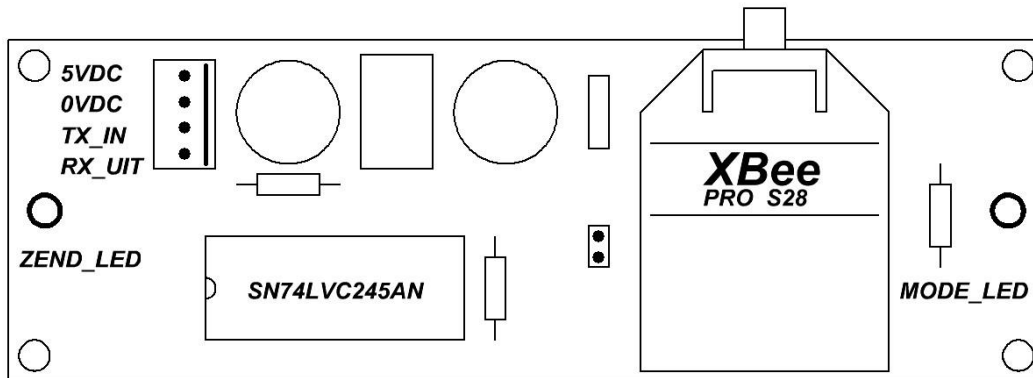
```
; DIGITALE UITGANG D_11 - AAN
If DATA_IN = 1 Then
  HRSOut 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,
    0XF2,0XFF,0XFE,0X02,0X50,0X31,0X05,0X29
  Print At 1,1,"D0=1"
  DelayMS TIJD
EndIf
```

En als er op drukknop twee UIT gedrukt wordt, wordt de code tien naar het controllerboard verzonden. De code wordt daar omgezet (door het basic programma) naar....

```
; DIGITALE UITGANG D_11 - UIT
If DATA_IN = 10 Then
  HRSOut 0X7E,0X00,0X10,0X17,0X01,0X00,0X13,0XA2,0X00,0X40,0X92,0XC0,
    0XF2,0XFF,0XFE,0X02,0X50,0X31,0X04,0X2A
  Print At 1,1,"D0=0"
  DelayMS TIJD
EndIf
```

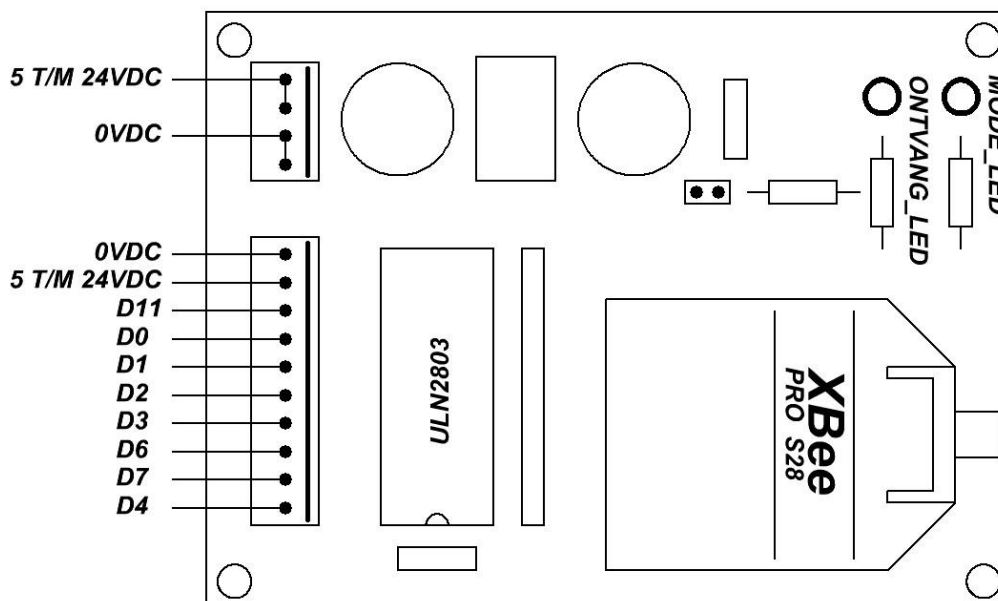
Tot zover de code in basic en profilab. We gaan nu de xbee modules behandelen.

Ik heb geen schema's gemaakt, maar heb wel de afbeeldingen van de printen, en de printen. Hieronder het zender gedeelte.



Hierop zijn de aansluitingen voor de voedingsspanning, en het zend en ontvang gedeelte te zien (TX_IN en RX_UIT). Er zitten ook twee leds op de print, één geeft de mode aan en één geeft het zenden aan. De mode led op het zend gedeelte knippert één maal per seconden.

Hier onder het ontvang gedeelte.

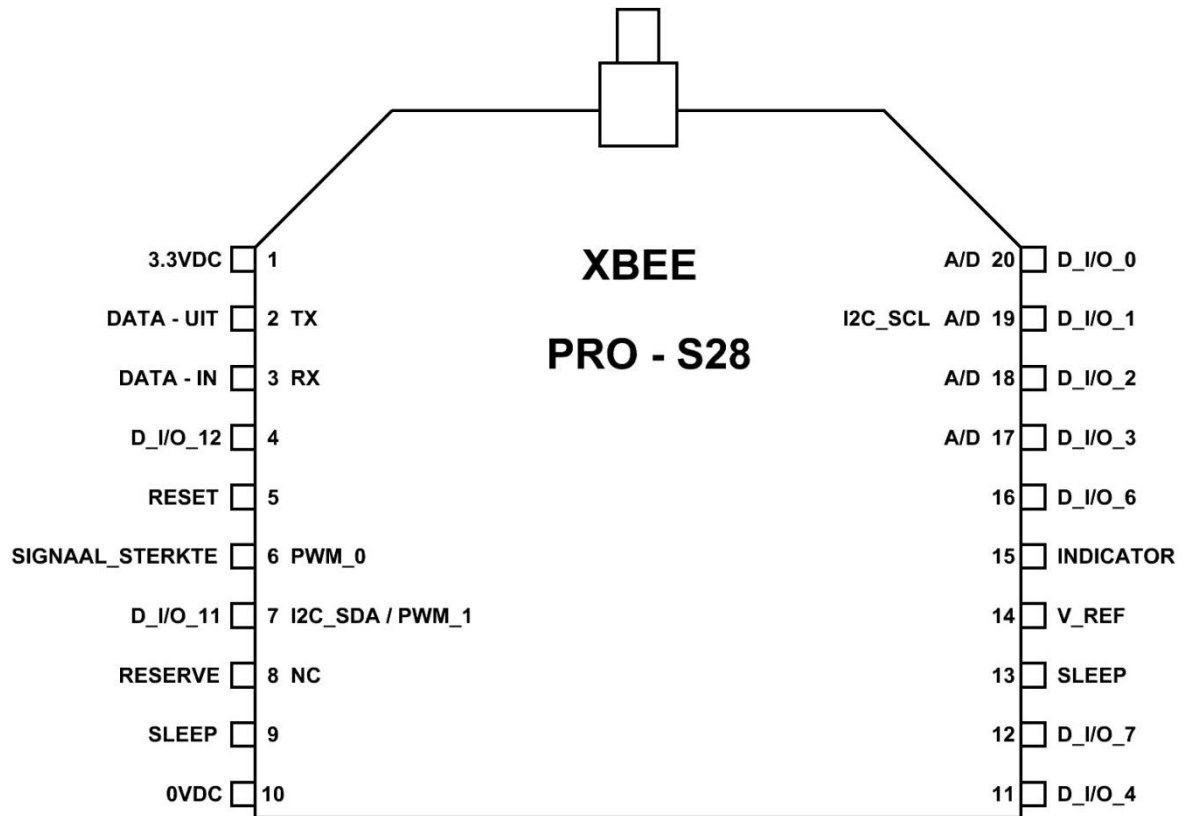


Het ontvanger gedeelte mag gevoed worden met een spanning tussen de 5 en 24VDC, die spanning ligt aan de te voeden belasting die aan de uitgangen zitten.

De pinnen van de Molex connector D11, D0, D1, D2, D3, D6, D7, en D4, komen overeen met de pinnen op de Xbee module. Tussen de 5 tot en met 24VDC en de D11 tot en met D4 wordt de belasting aangesloten. De belasting wordt ook hier weer naar de GND getrokken door de ULN2803. Ook hier staan weer twee leds op, één voor het te ontvangen signaal en één voor de mode.

De mode led voor de ontvanger knippert twee maal per seconden.
 Die leds gaan knipperen als de Xbee modules contact hebben met elkaar, dit gebeurt na een paar seconden als er spanning op de modules gezet wordt.

Hieronder de module van de Xbee.

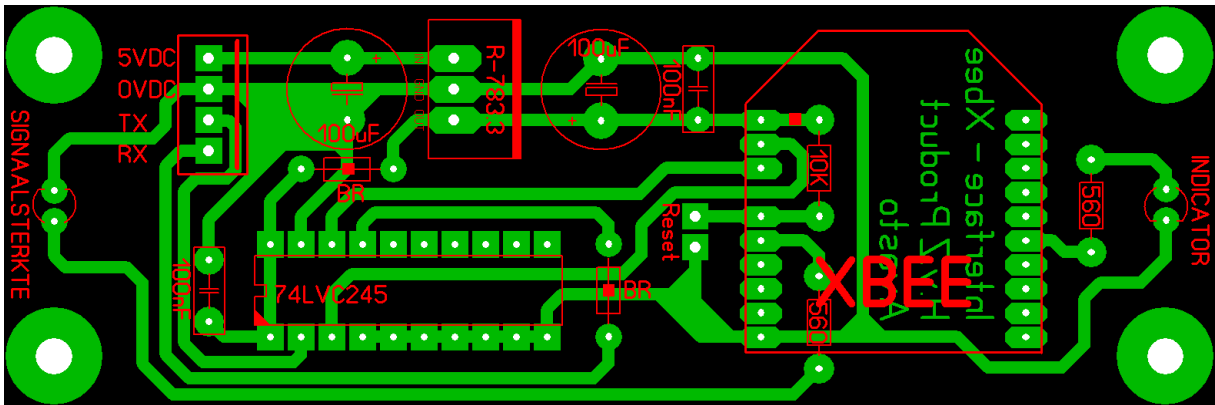


Hier staan alle pinnen van de Xbee module op, met de beschrijving erbij van de pinnen. De Xbee modules werken op een spanning van 3.3VDC. Op de zender en ontvanger print staan dan ook spanningsregelaars die de 5VDC voedingsspanning omzetten naar 3.3VDC. De regelaars van het geschakelde type. Zoals je kan zien zijn pin_1 en pin_10 voor de voeding van de module, pin_2 en pin_3 zijn voor de seriële data overdracht, pin_6 is voor de ontvang en zend led, pin_15 is voor de mode led van de zender en ontvanger, en op de rest van de pinnen zitten in of uitgangen.

De XBEE – PRO – S28 is een normaal type Xbee module, deze kan alleen maar geconfigureerd worden via XCTU. Maar daarover later meer. Er zijn ook types die geprogrammeerd kunnen worden, daar kan men nog meer mee. Maar daar heb je dan ook software voor nodig.

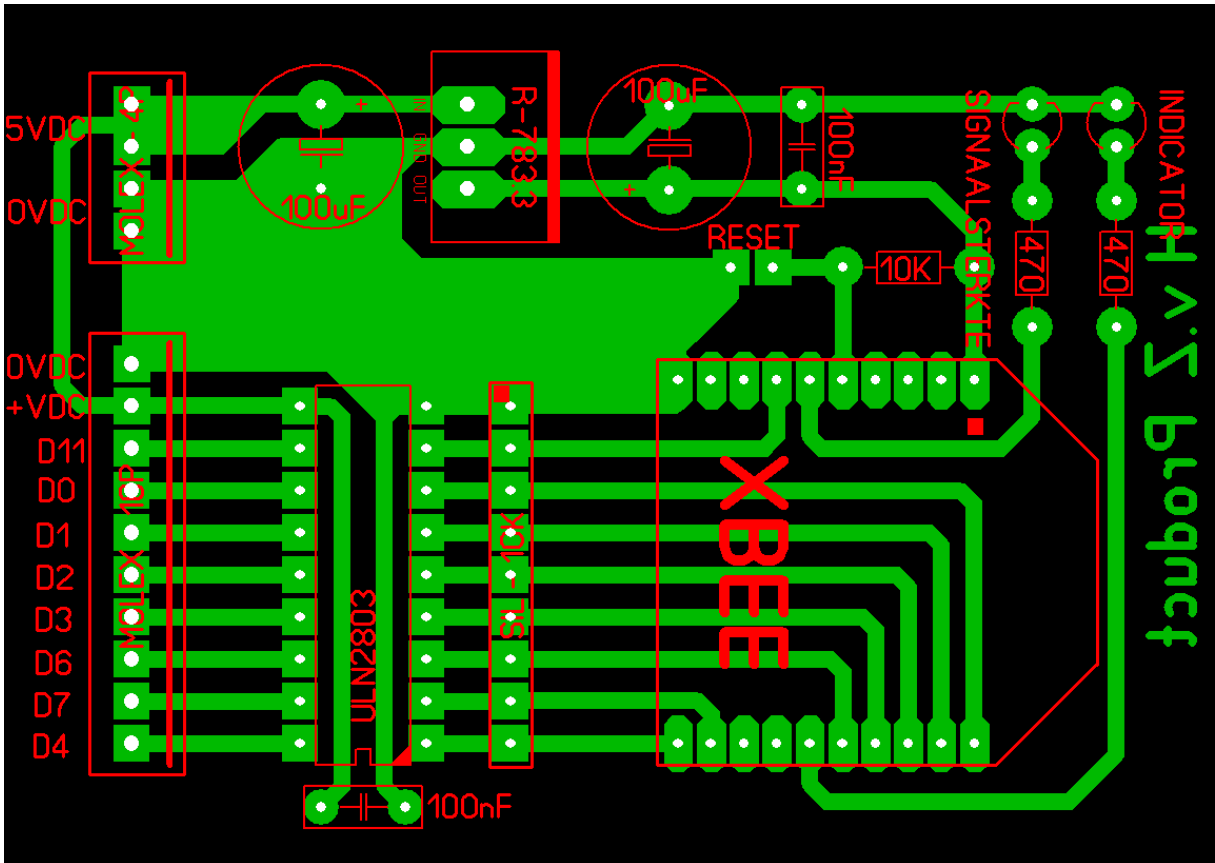
Er zijn ook weer printen voor de Xbee modules gemaakt. Zie Blz. 120.

Hieronder de print voor het zendergedeelte.



De afmeting van de print is 100 X 33mm.

Hieronder de print voor het ontvangergedeelte.

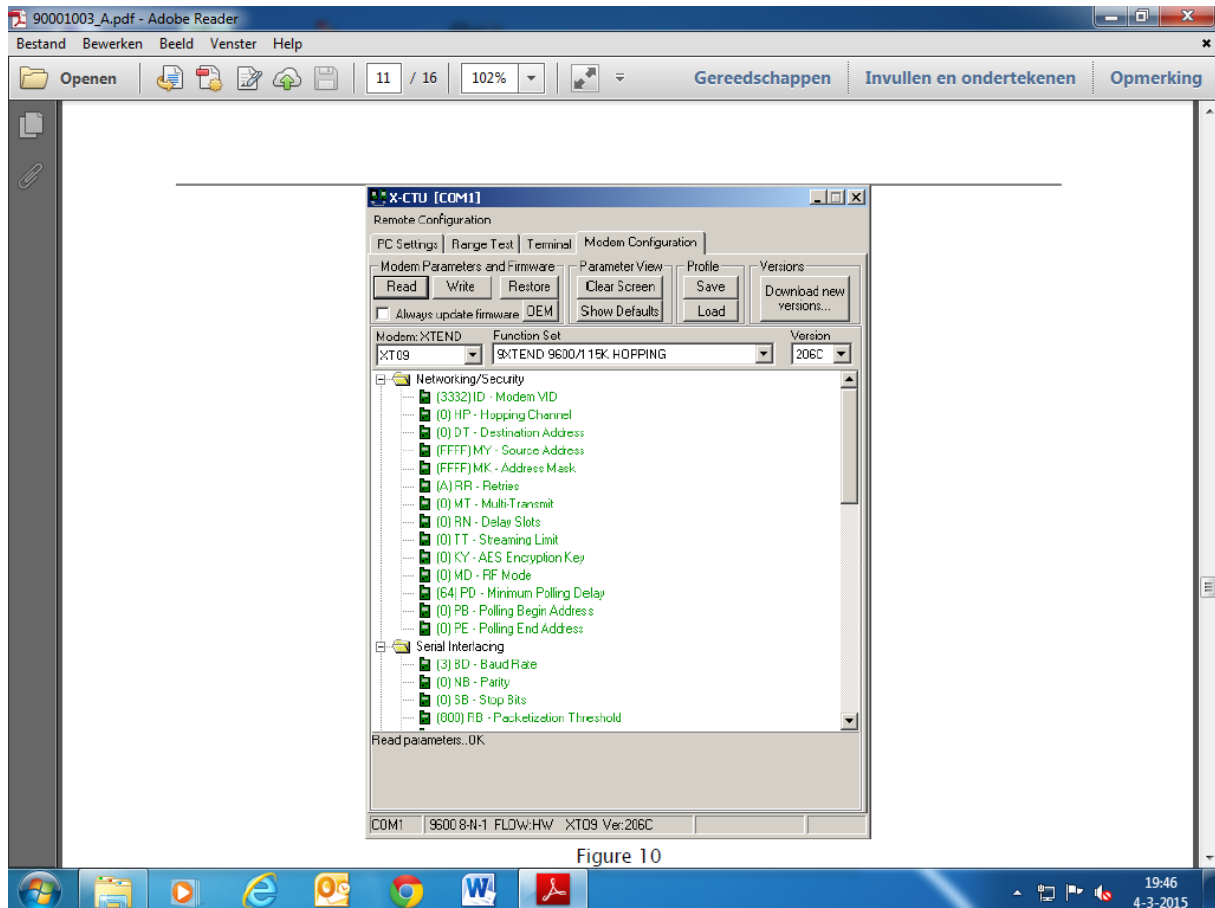


De afmeting van de print is 73 X 53mm.

We gaan nu de Xbee module bekijken.

Je kan het beste de datasheet van de Xbee van internet halen, daar staat heel veel nuttige info in over de Xbee. Je kan bijv. zoeken onder PDF XBEE - PRO, dan krijg je de uitgebreide datasheet van de Xbee.

Om de Xbee te configureren heb je een RS232 interface board nodig en het programma X – CTU, dat is te downloaden bij www.digi.com Dat programma ziet er zo uit.



Met dit programma kan je de instellingen van de Xbee module veranderen en aanpassen.

Je kan de poort instellingen veranderen of het bijv. in of uitgangen moeten zijn, het PANID kan hier ingevuld worden en allerlei andere instellingen kunnen gedaan worden.

Je kan ook bijv. tekst van de ene naar de andere pc draadloos verzenden, je hebt dan wel twee RS232 interfaces nodig.

Met het programma DIGI API FRAME MAKER kan je de code maken die bijv. door een controller verzonden kan worden, dat is ook de code die we in het basic programma gebruikt hebben.

Dat kan je ook vinden in de datasheet van de Xbee module. Op blz. 122 heb je een voorbeeld van het programma.



In de velden kan je de gegevens invoeren, de uitleg staat achter de velden. De gegenereerde code ziet er dan bijv. zo uit.

Voorbeeld api frame maker.

```

FRAME_TYPE      =      0X17          REMOTE      =      AT COMMAND
API_MODE        =      API – 1
TYPE            =      ALL_FRAMES
LENGTE         =      0010
API             =      17
FRAME_ID        =      01
64_DEST_ADRES  =      000000000000FFFF
16_DEST_ADRES  =      FFFE
CMD_OPTION      =      02
AT_CMD         =      D4 = POORT_4
AT_CMD_DATA    =      5 = POORT_UIT   4 = POORT_AAN
CHECKSUM        =      6D
PACKET         =      7E 00 10 17 01 00 00 00 00 00 00 FF FF FF FE 02 44 34 05 6D
  
```

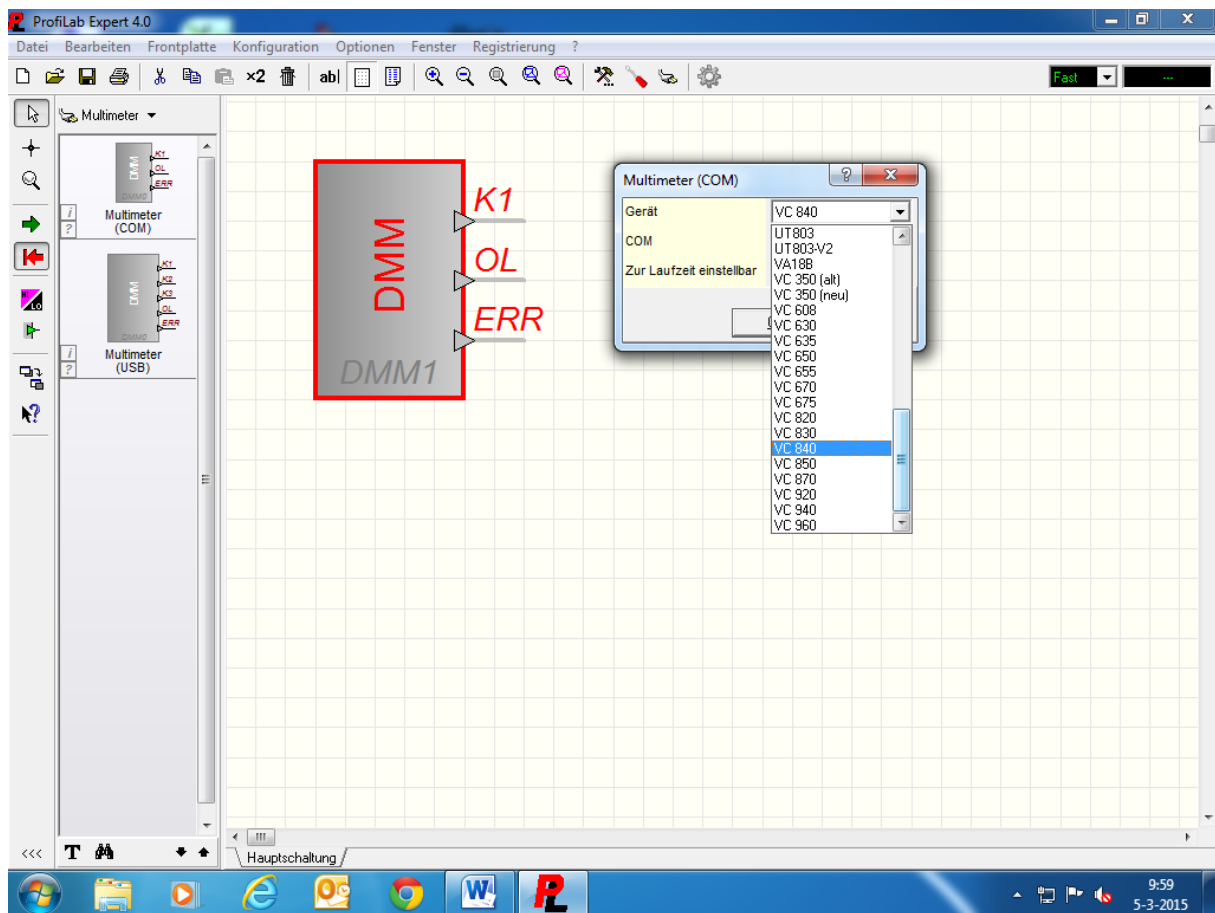
Die code moeten we wel een beetje aanpassen, daar moeten we overal 0X voor zetten. En dan krijg je de code die hier onder staat en dat vul je dan in, in het basic programma.

```
0X7E, 0X00, 0X10, 0X17, 0X01, 0X00, 0X00, 0X00, 0X00, 0X00, 0X00, 0XFF, 0XFF, 0XFF,  
0XFE, 0X02, 0X44, 0X34, 0X05, 0X6D
```

Tot zover het voorbeeld met de Xbee modules, ik hoop dat het een beetje duidelijk is zo.

Hoofdstuk 7: Werken met een digitale multimeter

Met ProfiLab expert 4.0 heb je de mogelijkheid om met een multimeter data binnen te halen en bijv. weg te schrijven. Er bevinden zich twee interface modules (één voor de com poort en één voor de usb poort) in het programma om een multimeter aan te koppelen. Zie voorbeeld hieronder.



Zelf gebruik ik de Voltcraft VC840 voor de metingen, maar er worden zo'n zestig verschillende meters ondersteund door ProfiLab. Hier een kleine opsomming BENNING MM10, DMM750, DT4000ZC, DVM345DI, M3610D, M3640D, M3650D, M3830, M3850M, M3860M, M384650CR, M4660, ME22, ME32, ME42, MS9150, MS9160, MS1280, MXD4660A, UT60E, UT61E, UT70B, UT70D, VC350, VC608, VC630, VC635, VC655, VC670, VC675, VC820, VC830, VC840, VC850, VC870, VC920, VC940, EN DE VC960. Op de meter zit een optische RS232 aansluiting, het signaal gaat via een kabel naar de RS232 poort van de pc. Daar kan het inkomende signaal bewerkt worden door ProfiLab.

Hieronder de multimeter van Voltcraft.



Dit is de meter die ik gebruik voor metingen via Profilab. Ik heb toen wel een gekalibreerde meter genomen, omdat er ook vaak aan het net gemeten wordt. Ik hang de meter nog al eens aan het net als schrijver, de data wordt dan in een grafiek gezet. Op deze manier kan je eenvoudig de netspanning voor één of twee dagen bekijken.

Ook gebruik ik de meter als temperatuur logger, om de temperatuur van iets langer in de gaten te houden.

Bij de meter zit ook een temperatuur meet probe, met de meter kan alleen in graden Celsius gemeten worden.

Om bijv. de stroom van een motor te meten gebruik ik een Stroomtang Voltcraft 160. Deze tang zet de stroom om naar een spanning. In dit geval is 5 millivolt één Amp. De tang is geschikt voor stromen tot 600A zowel voor AC als DC.

Op Blz. 125 zie je de seriële kabel die bij de meter zit.

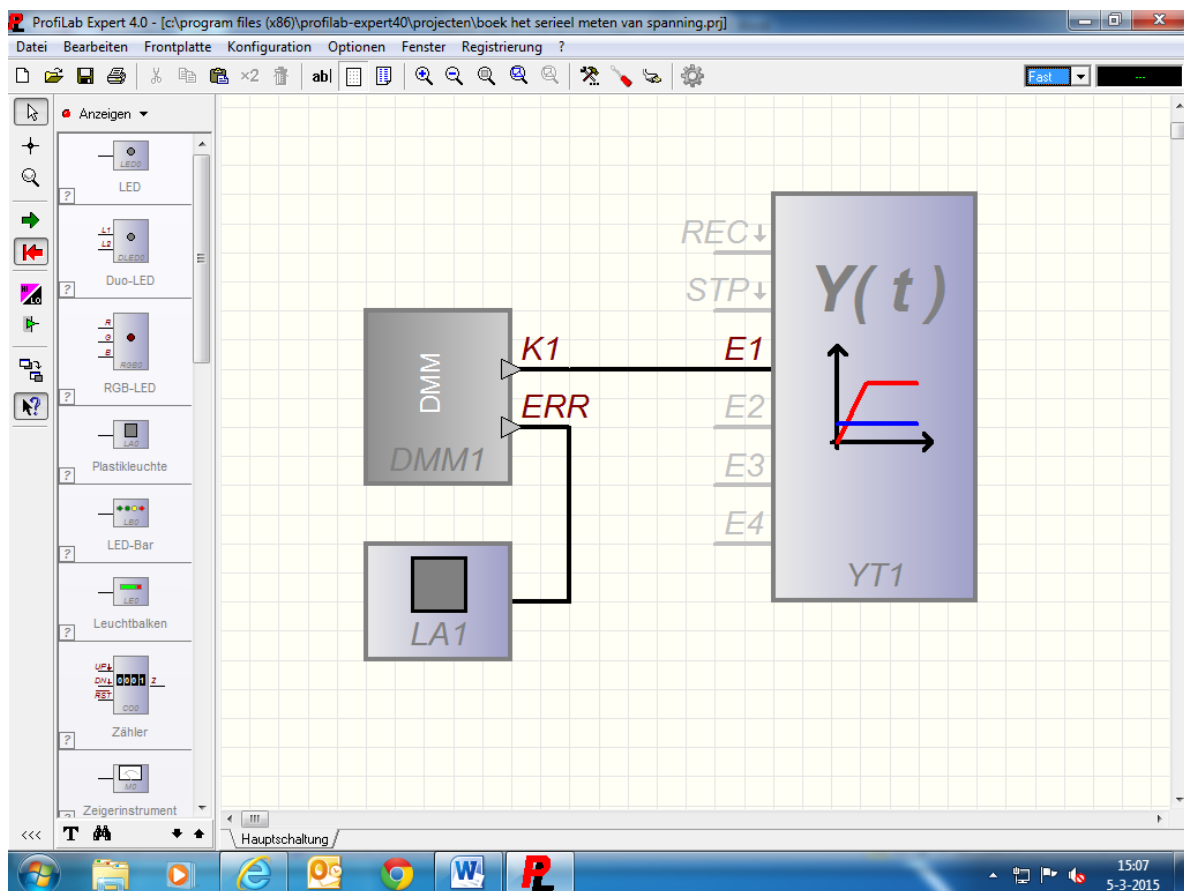
Hieronder de seriële kabel.



Tot zover over de meter, we gaan een paar voorbeelden maken met Profilab.

7.1: Het serieel meten van spanning

We gaan de netspanning meten hier thuis. Hieronder het programma in Profilab.

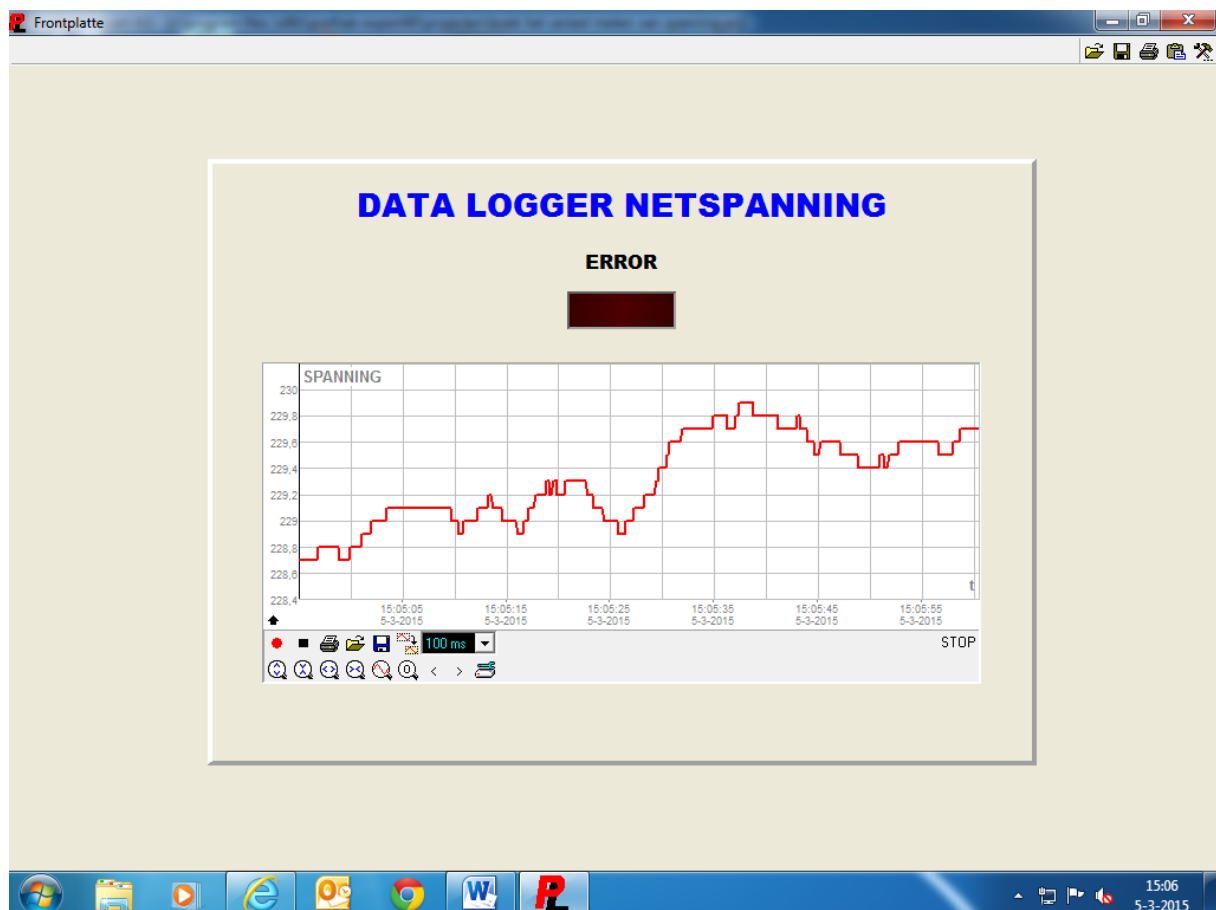


Zoals je kan zien is er niet veel voor nodig om een meting te doen. Ik heb de digitale multimeter module DMM1 gebruikt, dit is de module voor de RS232 poort. Onder eigenschappen selecteer je de meter die je gebruikt, in dit geval dus de VC840. Daar moet je ook de com poort selecteren, in dit geval is dat com_5. De K1 uitgang van de digitale multimeter module DMM1 stuurt een analoog signaal uit, deze wordt aangesloten op de E1 ingang van de analoge schrijver YT1. Op de error uitgang van de digitale multimeter module DMM1 sluiten we een lamp aan, op deze lamp is te zien of de communicatie in orde is. Als de lamp uit is dan is de communicatie in orde, als de lamp brand dan is er geen communicatie met de seriële poort. Dit kan bijv. komen omdat er geen com poort ingegeven is, of er is bijv. een verkeerde multimeter opgegeven.

We hebben nu voor een schrijver gekozen, maar dat kan ook bijv. een tabel module zijn. De data van een schrijver kan niet naar bijv. Excel of Word geschreven worden, maar wel naar een map die bijv. onder mijn documenten staat. Dit kan je doen door op de diskette knop te klikken, en aan te geven in welke map het moet komen.

Je kan de meetgegevens laden door het programma op te starten en op de map openen te klikken, en naar de map te gaan waar de gegevens opgeslagen zijn.

Hieronder de frontplaat van de spanning logger.



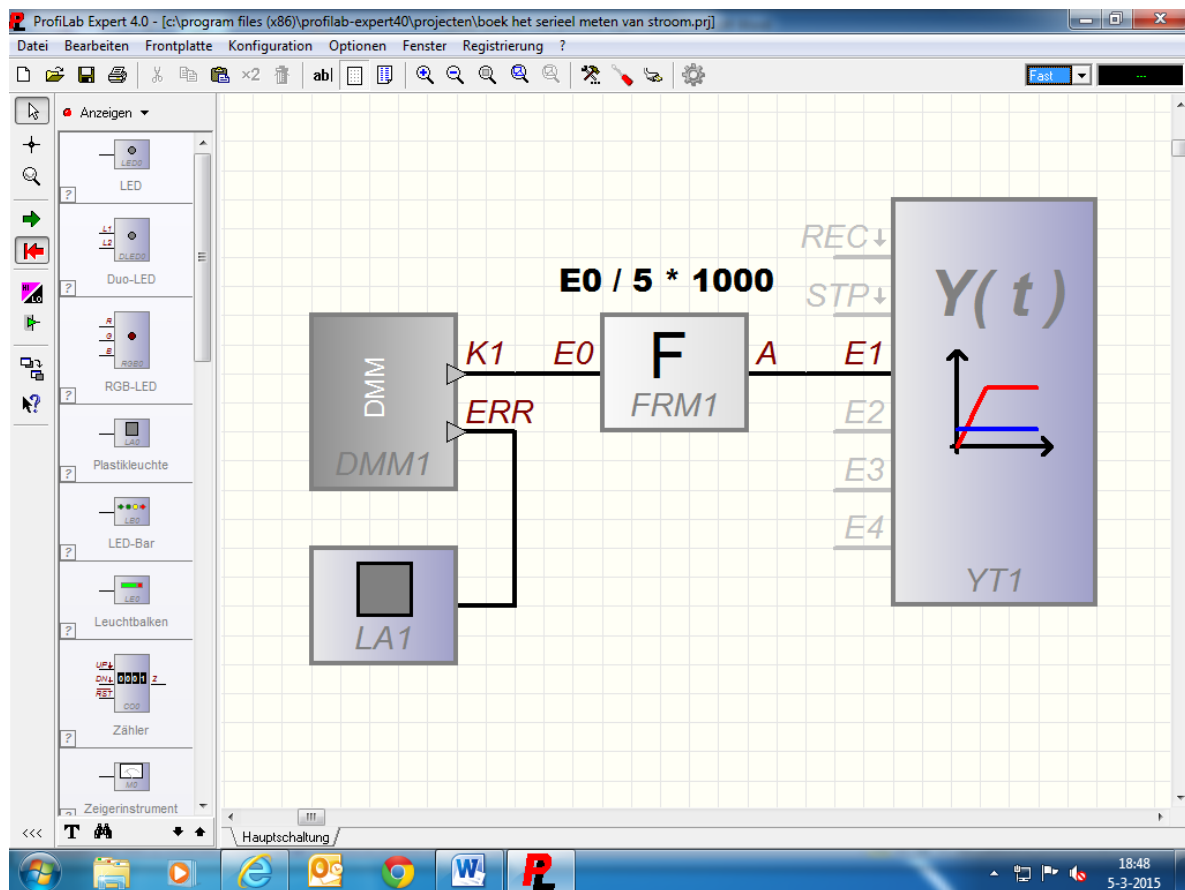
Op Blz. 126 kan je de gemeten data zien van het net. Zoals je kan zien fluctueert de spanning tussen +/- 228 en 229VAC. Er zijn dus niet echt grote spanning dippen te zien. Je kan ook zien dat de datum en de tijd onder in het beeld mee loopt, je kan dus precies nagaan Wanneer wat gebeurt.

Verder valt hier eigenlijk niet veel over te vertellen, het wijst voor zich lijkt mij.

7.2: Het serieel meten van stroom

We gaan nu een voorbeeld maken om de stroom van bijv. een elektromotor te meten. Daar kunnen we de zelfde modules voor gebruiken als voor de spanning meting, de enige module die er bij komt is een formule module. De frontplaat wordt eigenlijk ook niet veel anders, alleen de tekst wordt aangepast.

We gaan nu het programma maken.

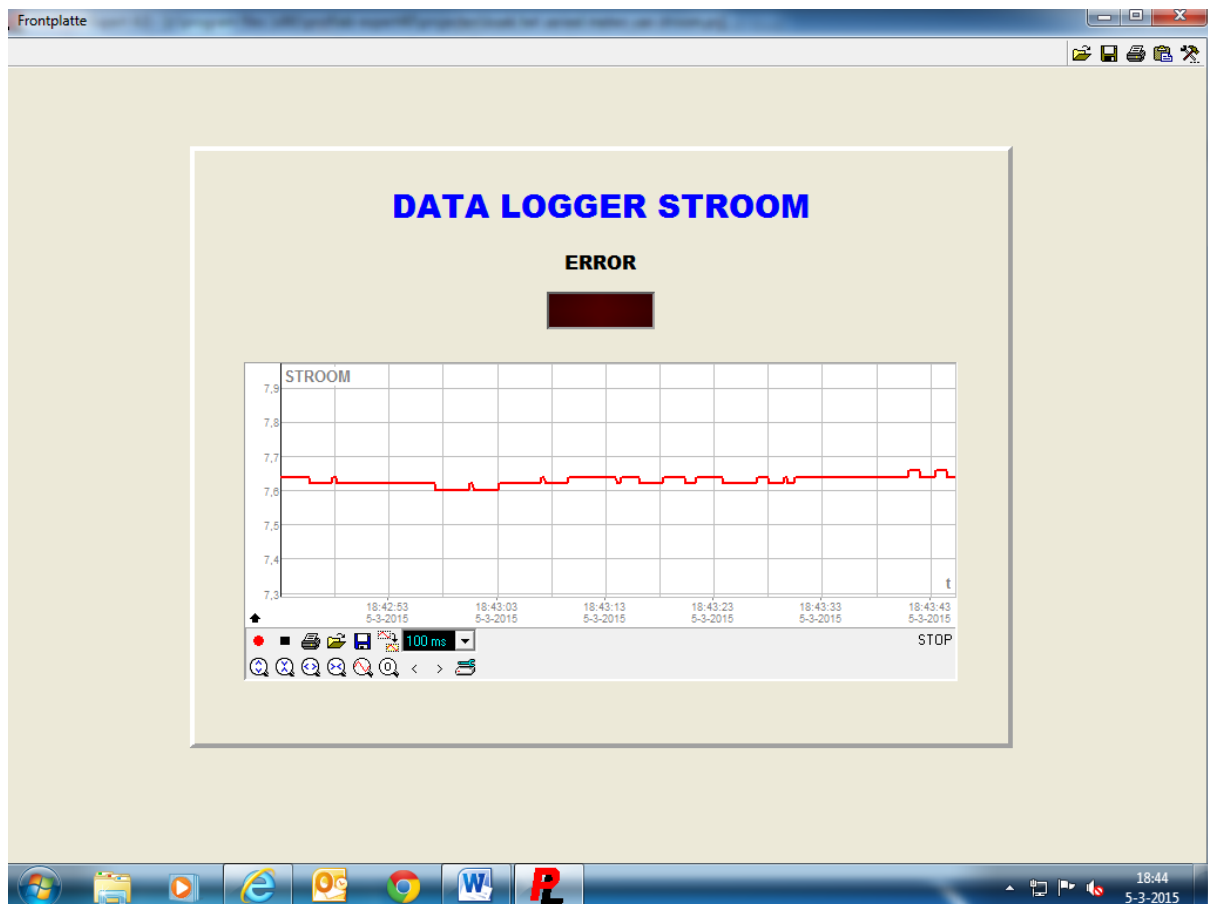


Zoals je kan zien is alleen de formule module er bij gekomen. De stroomtang geeft een spanning af van 5mV per Ampère. Met de formule $E0 / 5 * 1000$ krijgen we dus $0.005 : 5 = 0.001 \times 1000 = 1$. Op deze manier zetten we de 5mV spanning om naar 1A.

Op Blz. 128 de stroom adapter van Voltcraft, het is niet helemaal het zelfde type maar dan heb je een idee wat er bedoeld wordt.



We gaan nu de frontplaat weer maken, zie hieronder.

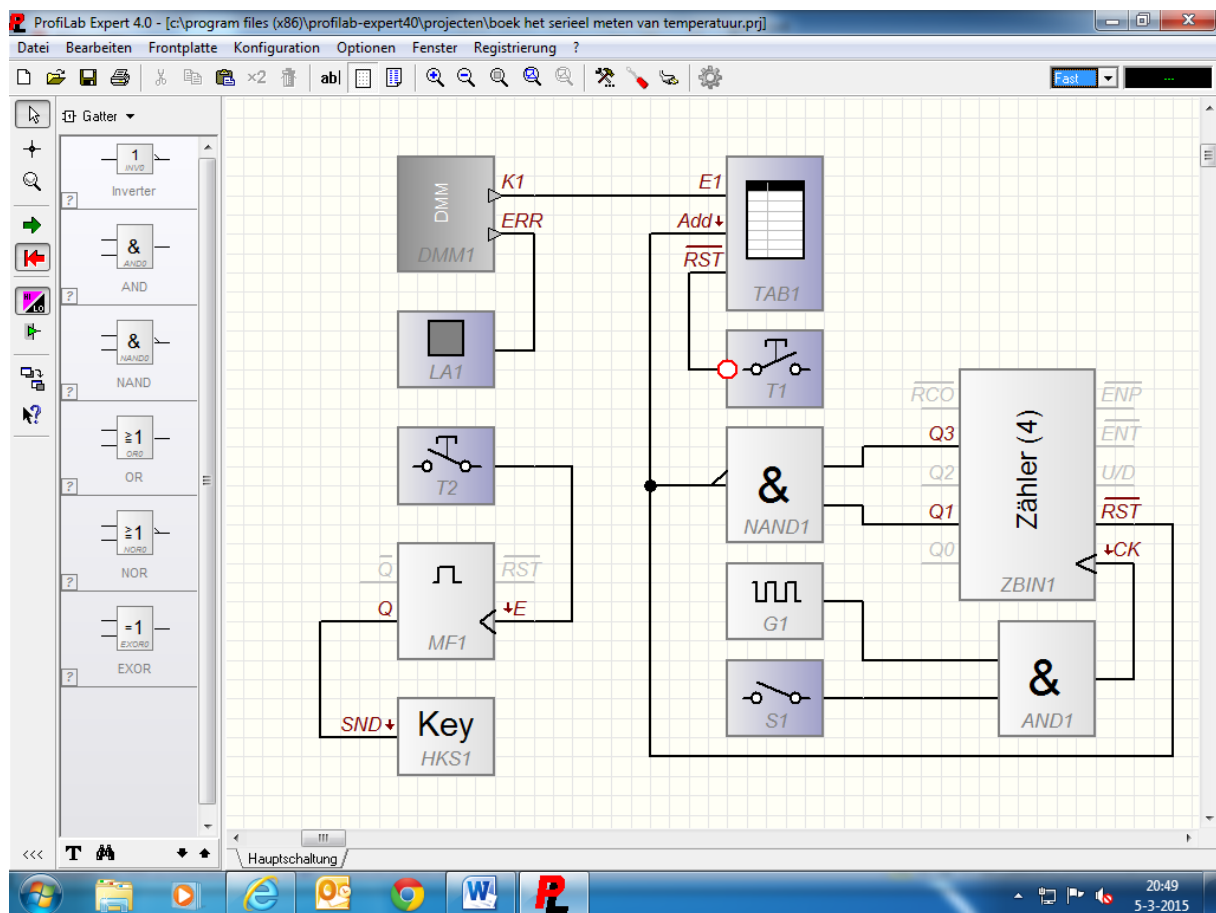


Zoals je kan zien is er niet veel veranderd, alleen een stukje tekst. Ik heb hier de stroom van een elektromotor gemeten. De motor is een 24VDC type, met een vermogen van 180W. Zoals je kan zien neemt de motor +/- 7.5 Ampère op. Ook deze data kan weer in een map gezet worden.

7.3: Het serieel meten van temperatuur

We gaan nu een voorbeeld van een temperatuur logger maken. We maken weer gebruik van de zelfde modules. Alleen gaan we de schrijver vervangen door een tabel module, de data die in de tabel module komt te staan kunnen we naar Word of Excel schrijven. Dat kan handig zijn als je bijvoorbeeld dagelijks een temperatuur van iets bij wilt of moet houden. Je kan dat bijvoorbeeld gebruiken voor een broedmachine, of een broeikas met planten, enz., enz. Deze metingen doen we nu met tussenkomst van een digitale multimeter, maar dat zou ook kunnen via het controllerboard die we besproken hebben. Op die manier kan je ook een data log systeem maken natuurlijk.

We gaan eerst weer het programma met Profilab maken. Zie hieronder.



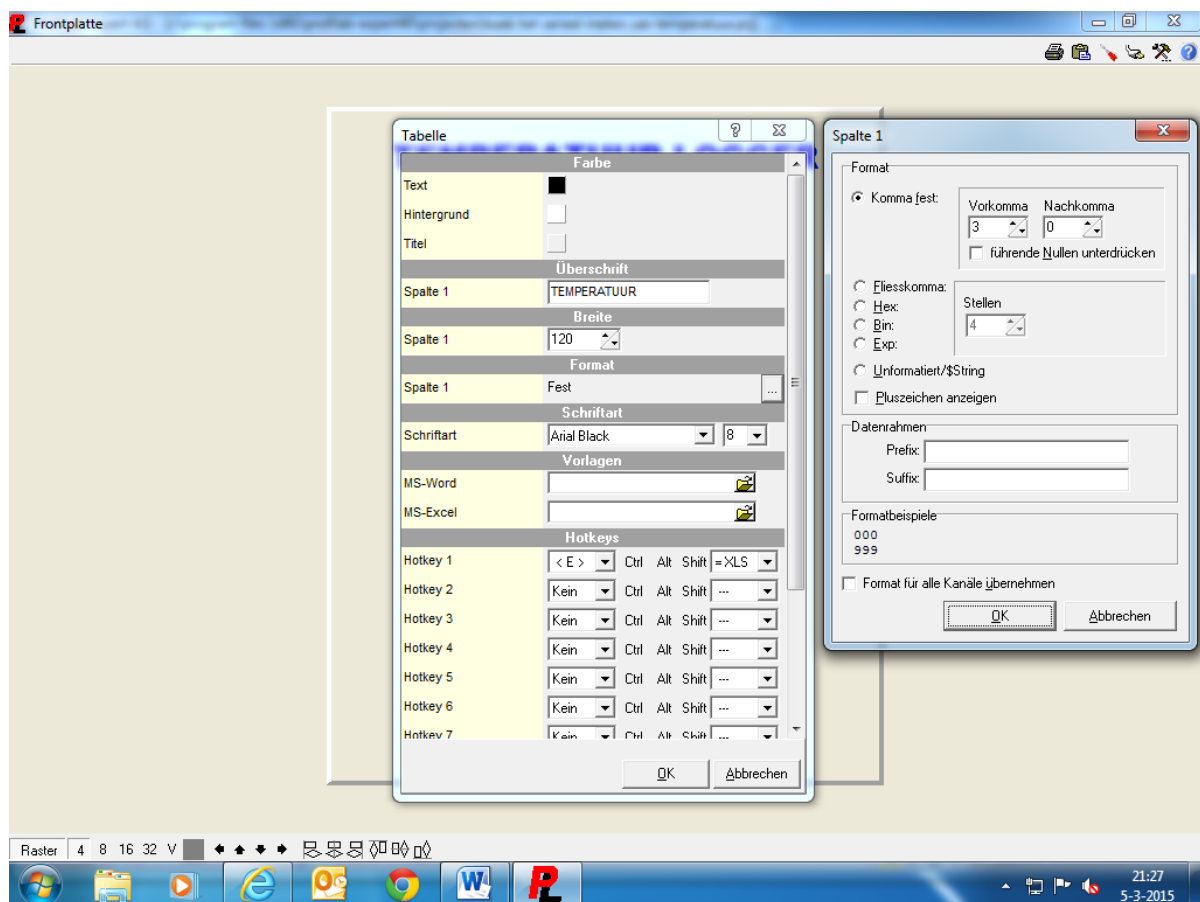
Hier kan je zien dat de schrijver vervangen is door de tabel module TAB1. Met de drukknop T1 kan de inhoud uit de tabel module verwijderd worden. Er is een teller module toegevoegd zodat de data iedere tien seconden geladen wordt, deze tijd kan aan gepast worden als men dat wil. Je kan bijvoorbeeld eens in de tien minuten, eens per half uur, of eens per uur data naar de tabel schrijven. Maar dan zal de teller veranderd moeten worden, je kan daar bijvoorbeeld ook een klok module neer zetten. Met de schakelaar S1 kan de data logger aan en uit gezet worden, de puls geveer G1 staat hier op één puls per seconde, 1Hz dus.

Met de drukknop T2 kan de data naar bijv. Word of Excel geschreven worden, ik heb in dit voorbeeld gekozen voor Excel. Achter de drukknop T2 staat een monoflop module, deze monoflop zorgt ervoor dat als de knop T1 ingedrukt wordt het puls signaal van de drukknop voor 0.5 seconden aanwezig blijft. Dat zorgt ervoor dat Excel goed opgestart wordt.

Achter de monoflop staat een hotkey module HKS1, die zetten we onder eigenschappen op <E>. Op die manier kan je de data zowel met drukknop T2 in Excel zetten, maar ook als je de E op het toetsenbord indrukt.

Om de data naar Excel te krijgen moeten we de tabel module TAB1 in stellen, ik ga dat laten zien aan de hand van een voorbeeld.

Zie hieronder.



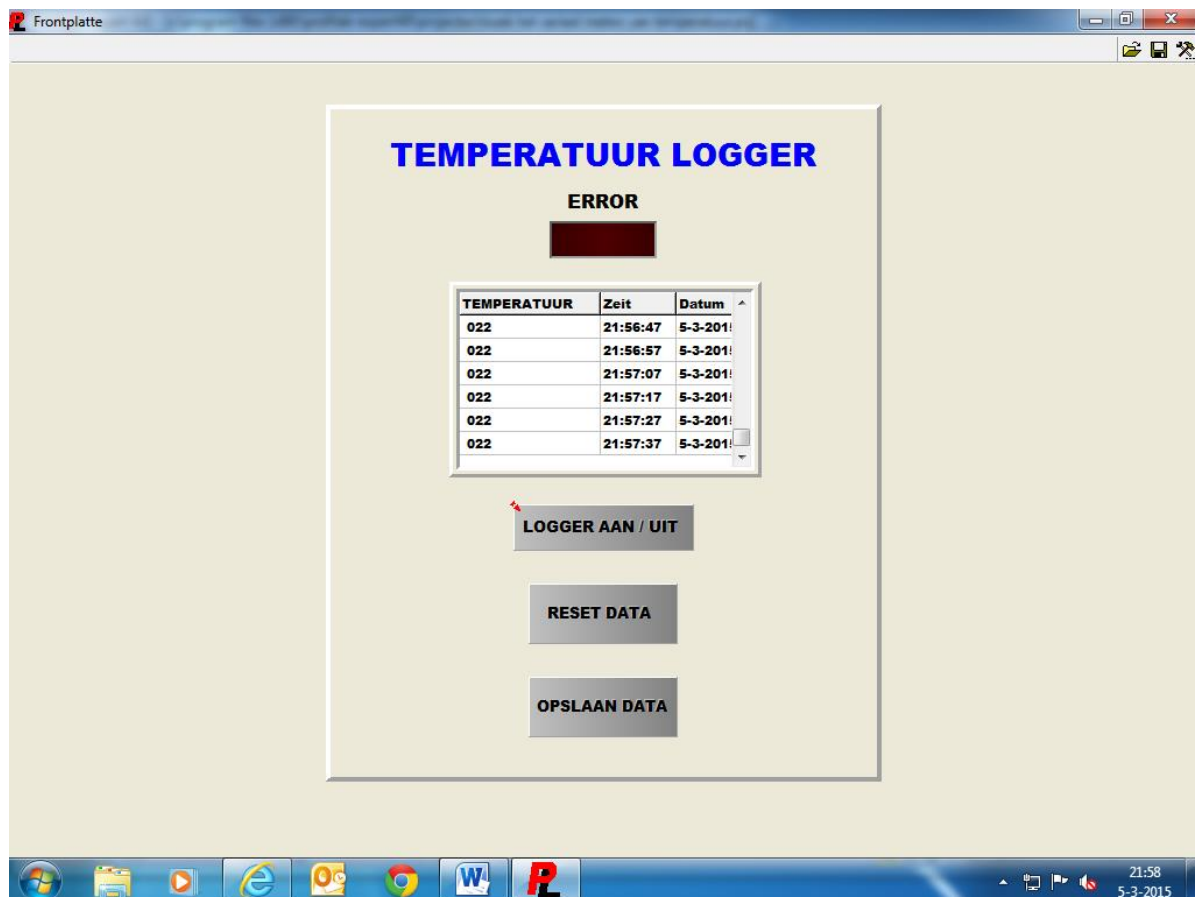
Onder eigenschappen van de tabel module kan je het één en ander instellen. We zetten daar de tekst TEMPERATUUR neer, de breedte van het eerste veld (waar temperatuur in komt te staan) zetten we op 120. Als we op het knopje met de drie puntjes klikken kunnen we het aantal digits van de tabel module instellen, deze zetten we op drie. Zoals je ziet heb je voor de instelling van het aantal digits aardig wat keuze, je hebt een keuze voor en na de komma, je kan voor Hex, Bin, Exp, en String als data kiezen.

De tekst waarmee de data in de tabel module komt te staan zetten we op Arial Black, en de grote van de tekst zetten we op acht.

In het eerste rijtje van de hotkeys zetten we de <E> neer (net zoals we gedaan hebben bij de hotkey module HKS1. Als je nu aan de rechterkant kijkt dan kan je zien dat je kan kiezen uit ADD, RST, CPY, XLS, DOC, en TXT. De eerste twee zijn hier niet van belang, maar de vier opvolgende wel. Met CPY kan je een copy maken van de data, met XLS stuur je de data naar Excel, met DOC stuur je de data naar een Word document, en met TXT stuur je de data naar een tekst document.

Wij hebben er voor gekozen om het naar Excel te sturen en daar op te slaan.

We gaan nu de frontplaat samen stellen. Zie hieronder.

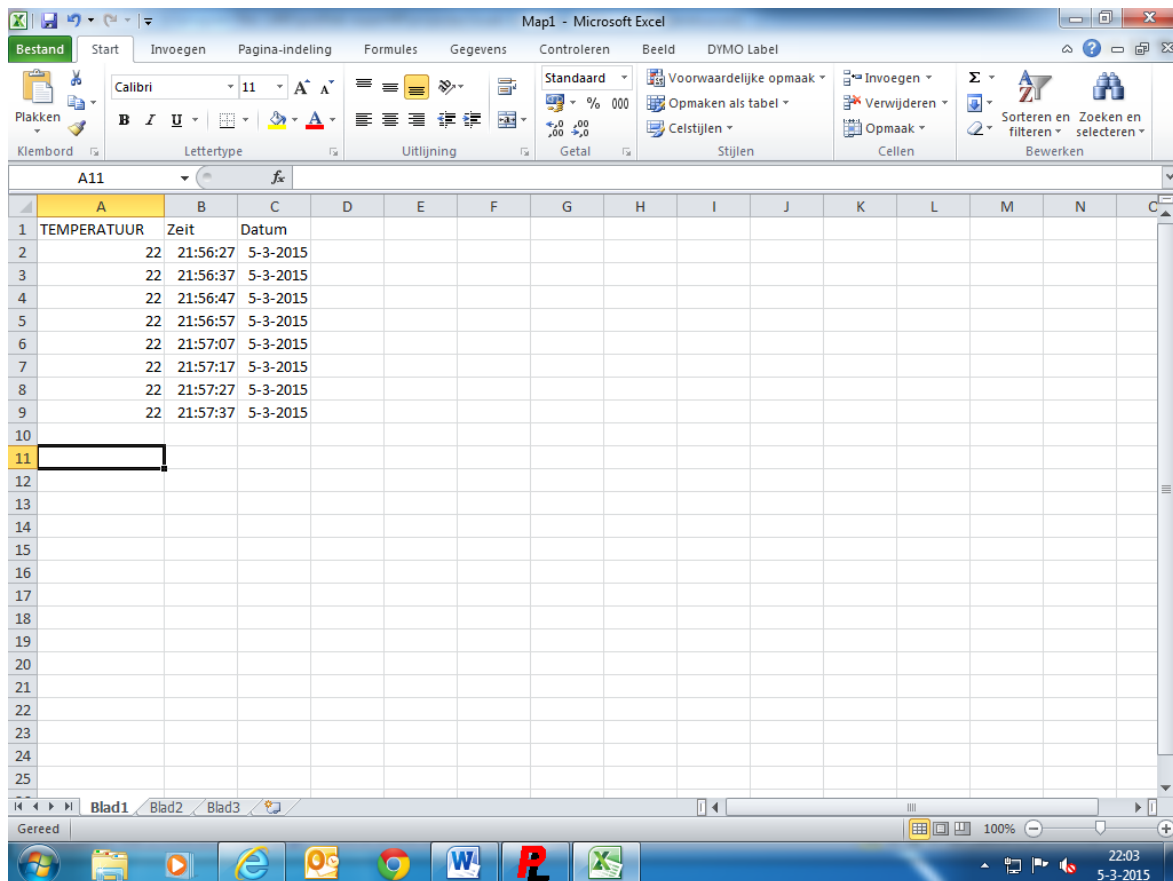


Op deze manier heb ik de frontplaat gemaakt, zoals je ziet heb ik hem even laten lopen en de temperatuur in de woonkamer gemeten. Je kan zien dat er iedere tien seconden een waarde wordt ingelezen, daarnaast staat de tijd en de datum. Met de knop logger aan / uit kan je de logger starten en stil zetten. Onder start / stop staat de reset knop, daarmee wordt de data gewist uit de tabel.

Met de knop opslaan data ga je naar Excel.

Hier naast zie je een voorbeeld hoe het in Excel komt te staan. Je ziet dat de temperatuur 22 graden aangeeft, met daar achter de tijd en de datum van de gemeten temperatuur. Op deze manier kan je de data opslaan.

Op Blz. 132 zie je het werkblad van Excel

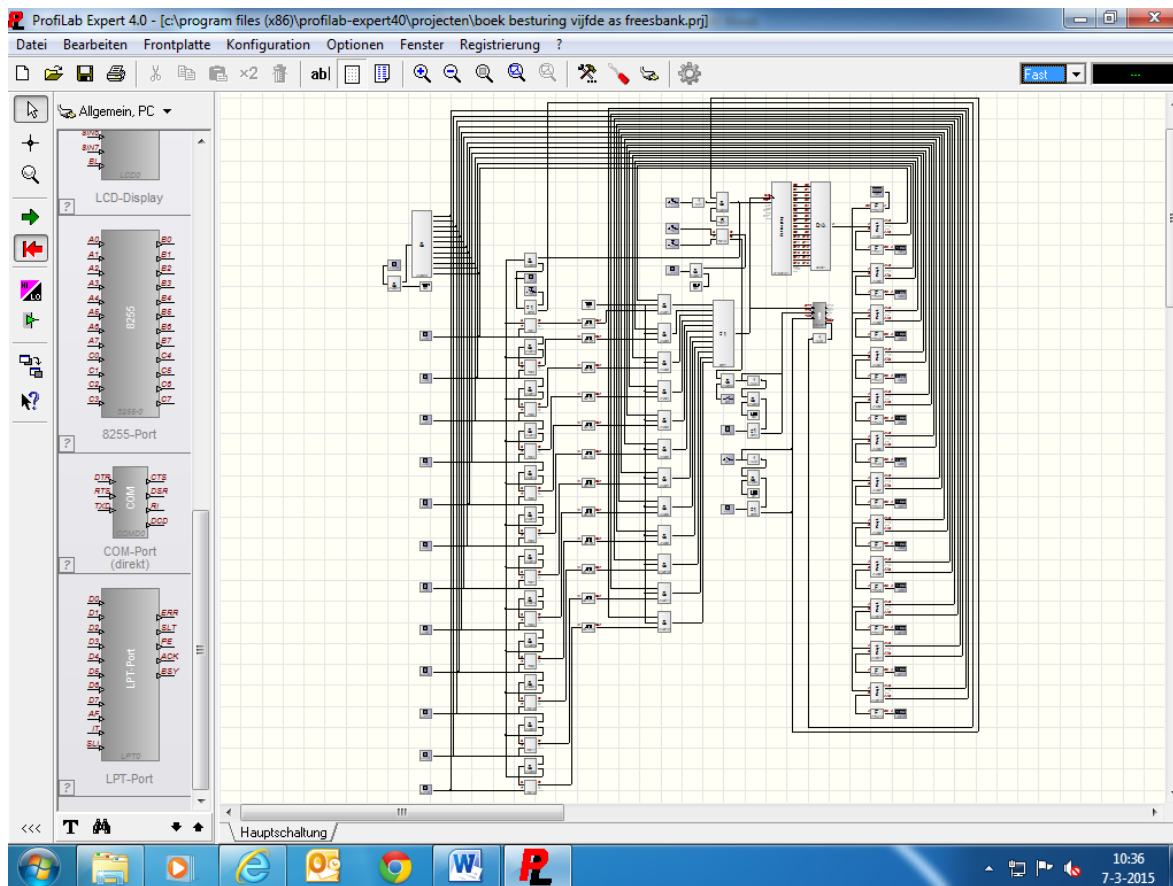


Hoofdstuk 8: Een paar praktijk voorbeelden

Ik ga nog wat praktijk voorbeelden laten zien. De voorbeelden zijn: een vijfde as voor een freesbank, een alarm melding op een schip, sprong test training voor bijv. handbal, korfbal, basketbal , enz., Doel training voor zaalvoetbal, circuittraining voor fitness. Dit zijn zomaar een paar voorbeelden die ik voor de praktijk gemaakt heb.

8.1: Vijfde as voor een freesbank

Op Blz. 133 zie je het werkblad met het programma staan.



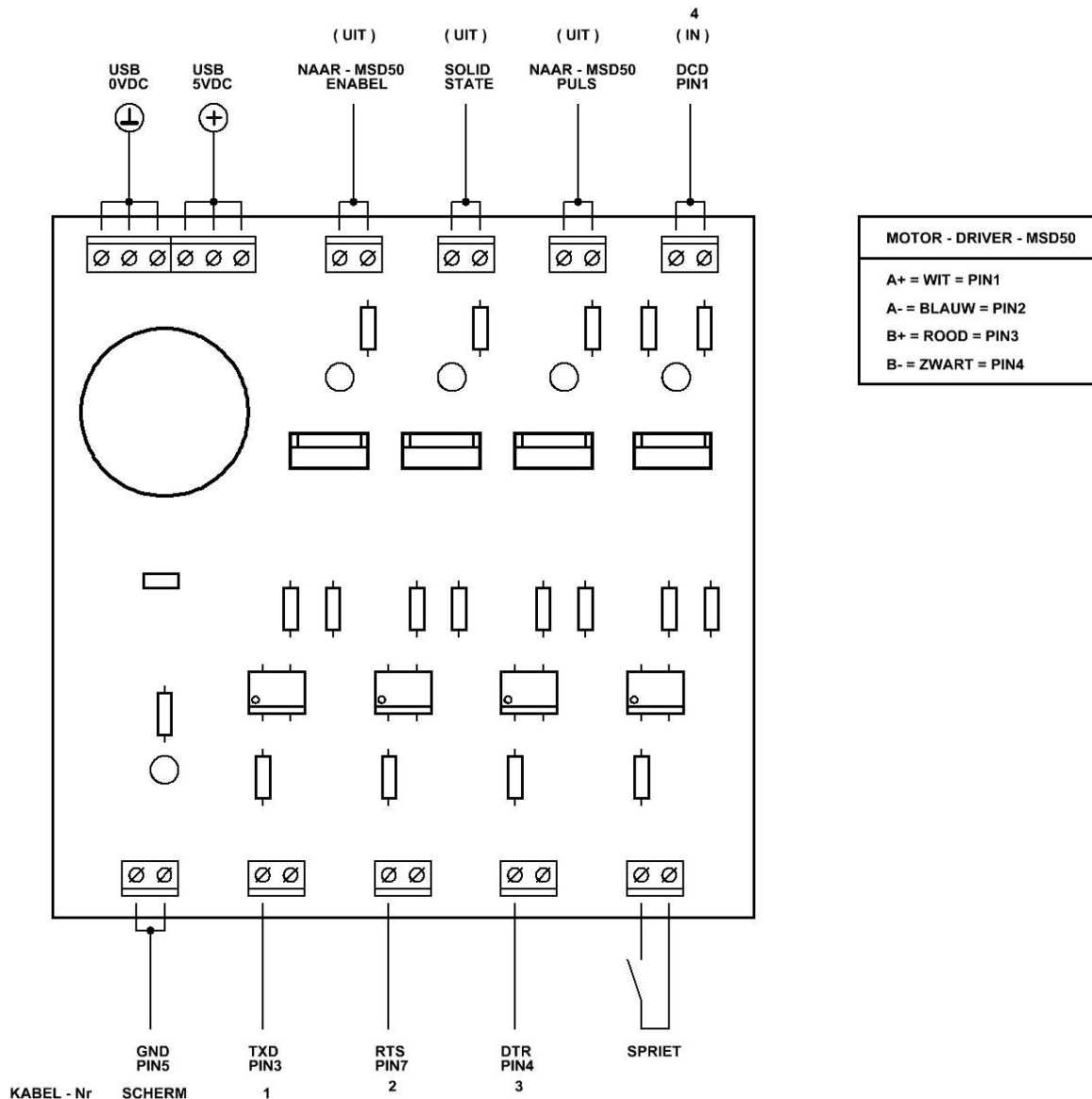
Voor de uitsturing van de data wordt de RS232 poort van de pc gebruikt, aan de poort hangt een interface die een stappenmotor driver aan stuurt. De signalen zijn de puls voor de driver, het vrijgave signaal voor de driver, het signaal voor de rem, en een ingang voor de spriet. De spriet wordt aangetikt door de frees kop van de freesbank als de klauw koppen moeten verdraaien. Dat doet de machine voordat het gereedschap gewisseld wordt.

De RS232 poort stuurt een signaal van 1KHz naar de stappenmotor driver, dit is de stapsgesnelheid voor de stappenmotor.

Op blz. 134 de interface van de besturing.

Hier is toen ook weer een print voor gemaakt, maar die kan ik niet tonen.

Interface vijfde as freesbank.

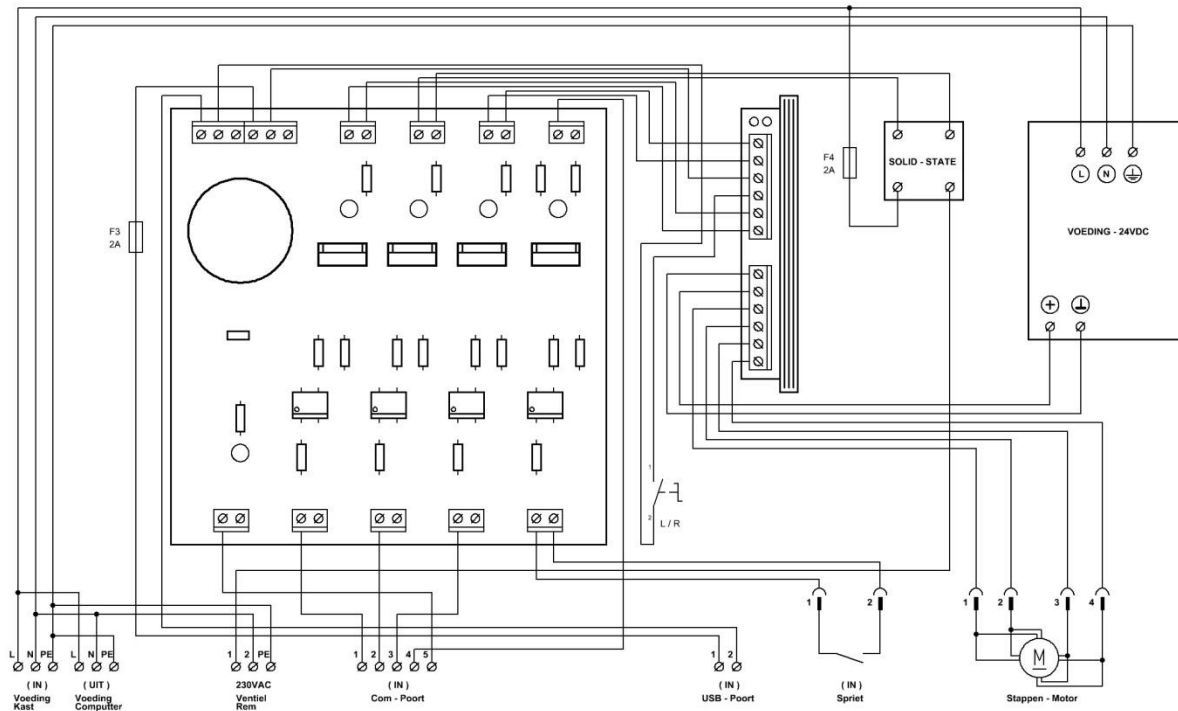


Hier zijn de aansluitingen te zien op de interface. De ingangen gaan via optocouplers, en de uitgangen worden door mosfets gestuurd.

Ook zijn de aansluitingen van de RS232 poort te zien, in dit geval wordt de RS232 poort als in en uitgangen gebruikt en niet voor de seriële overdracht van signalen. De DTR, RTS, en TXD pinnen van de RS232 poort kunnen als uitgangen gebruikt worden, en de CTS, DSR, RI, en DCD pinnen kunnen als ingang gebruikt worden.

Nog even iets over de RS232 poort, deze is te vinden onder hardware, algemene pc. Daar kan je verschillende poorten vinden, bijv. TCP, JOYSTICK, GAME PAD, COM POORT, en LPT POORT. De pinnen van deze poorten zijn direct te benaderen via Profilab.

Hieronder het aansluitschema.

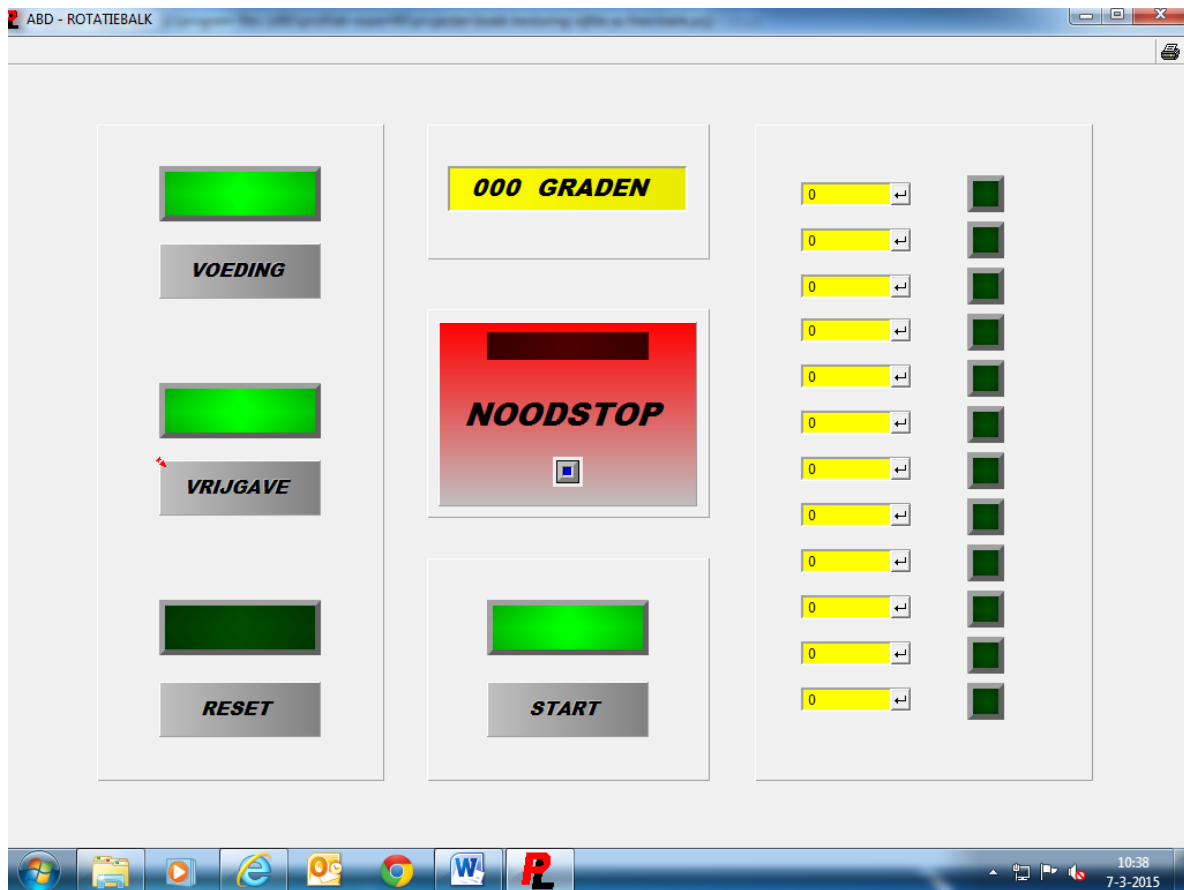


Dit is het totale plaatje van de besturing, zo kan je op eenvoudige wijze een besturing maken via de pc. Normaal ben ik er geen voorstander van om direct iets vanuit de pc aan te sturen, omdat je altijd met interrupts van de software kan krijgen. Maar in dit geval maakte dat niet veel uit, omdat de snelheid niet van belang is.

Normaal doe ik zo iets via een controller, zoals ik in de eerdere voorbeelden heb laten zien. Dan stuur je alleen maar commando's vanuit de pc, en de rest wordt afgehandeld door de controller.

De stappenmotor driver is van het type MSD50, dit is een microstep driver. Met deze driver kan je kiezen uit 1/2, 1/4, 1/8, 1/16, 1/32, 1/64, en 1/128 stap. Dat houdt dus in; stel dat je een motor hebt die 200 stappen per omwenteling maakt zou je die motor nu met bijv. 1/32 stap aansturen, dan maakt de motor geen 200 stappen per omwenteling maar $32 \times 200 = 6400$ stappen per omwenteling. Dan wordt de motor dus ineens een heel stuk nauwkeuriger. Bij 200 stappen is de verdraaiing van de motor dus $360 : 200 = 1.8$ graden per stap, en bij 6400 stappen is dat $360 : 6400 = 0.05625$ graden per stap. Dat scheelt dus nogal iets. Er kleeft ook weer een nadeel aan, de stapfrequentie moet ook een stuk hoger liggen om een beetje fatsoenlijk toerental te halen. En ook het geleverde koppel wordt er iets lager door.

Op blz. 136 de frontplaat van de regeling.



Dit is de indeling van de frontplaat. De werking is als volgt, de drukknoppen VOEDING, en VRIJGAVE zijn twee voorwaarden om het systeem op te starten. Als daar de lamp niet boven brand doet het systeem helemaal niets. De knop VOEDING heeft in dit geval niets met de voedingsspanning van het systeem te maken, dat betekent in dit geval de voeding van de as. Daar zit de vrijgave van de stappenmotor driver op. Met de VRIJGAVE knop wordt het rem systeem van de klauwkoppen vrijgegeven. Op de klauwkoppen zit een hydraulisch remsysteem, de klauwkoppen worden op de rem gehouden tijdens het boren of frezen. Als de klauwkoppen moeten verdraaien wordt de rem gelost.

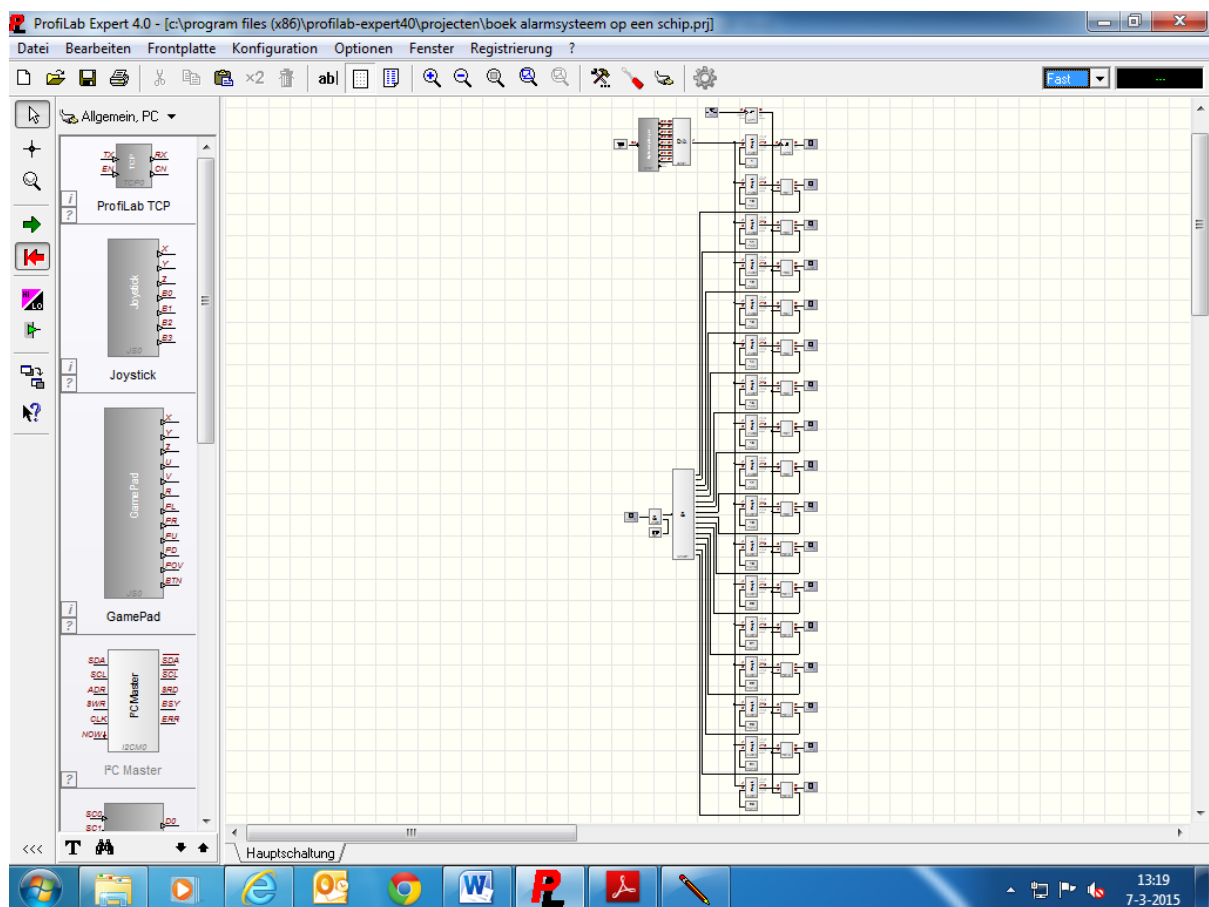
Het systeem kan zowel door de START knop gestart worden, als door de spriet die in de freesbank zit. Rechts zie je een aantal velden waar je het aantal graden van de verdraaiing van de klauwkoppen in kan stellen, als het aantal graden is bereikt gaat de lamp naast het veld branden en stoppen de klauwkoppen met draaien. Het aantal verdraaide graden wordt ook getoond in het display boven aan de frontplaat, zo is er dus altijd een visuele controle mogelijk.

Je kan dus op twaalf velden het aantal graden ingegeven, dat kan variëren van 1 graad tot 360 graden.

Dan hebben we nog de noodstop, de noodstop haalt het vrijgave signaal van de stappenmotor driver en zorgt ervoor dat er geen enkele knop meer bediend kan worden. Er is ook nog een noodstop op de kast aanwezig, deze noodstop schakelt het hele circuit uit.

8.2: Alarm melding voor een schip

Dit is een voorbeeld van een alarm systeem op een schip.



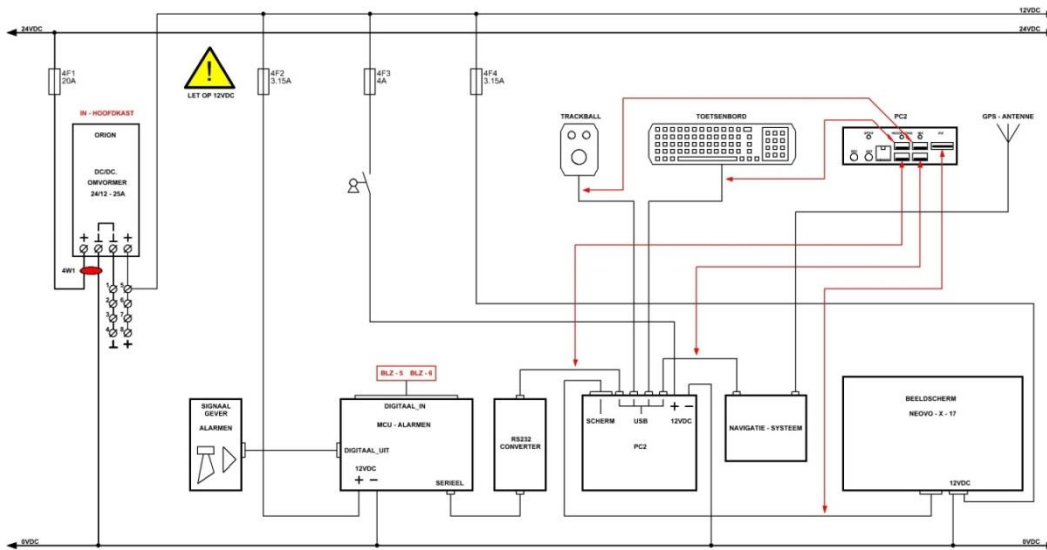
Dit is het complete programma, er helaas niet zoveel van te zien. Helemaal boven in staat de seriële module, deze module haalt de alarm signalen binnen via de pc. Als er een alarm is, dan gaat er een signaal af en er gaat een lamp knipperen op de frontplaat. De data die binnen komt van het controller board is; 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, en 1. Waarde 10 is brandmelding voorpiek, 11 is brandmelding slaapkamer, 12 is brandmelding woonkamer, 13 is brandmelding stuurhut, 14 is brandmelding machinekamer, 15 is bilge alarm voorpiek, 16 is bilge alarm midscheeps, 17 is bilge alarm machinekamer, 18 is watertank voor, 19 is watertank achter, 20 is vuilwatertank, 21 is dieseltank generator, 22 is dieseltank hoofdmotor, 23 is reserve-1, 24 is reserve-2, 24 is reserve-3, en 1 is het controle signaal voor de seriële poort.

De waarde die binnen komt wordt vergeleken met een waarde die in een vaste waarde module staat, (dit zijn modules die we in andere voorbeelden gebruikt hebben) als één van de waardes overeen komt dan volgt er een actie op de frontplaat.

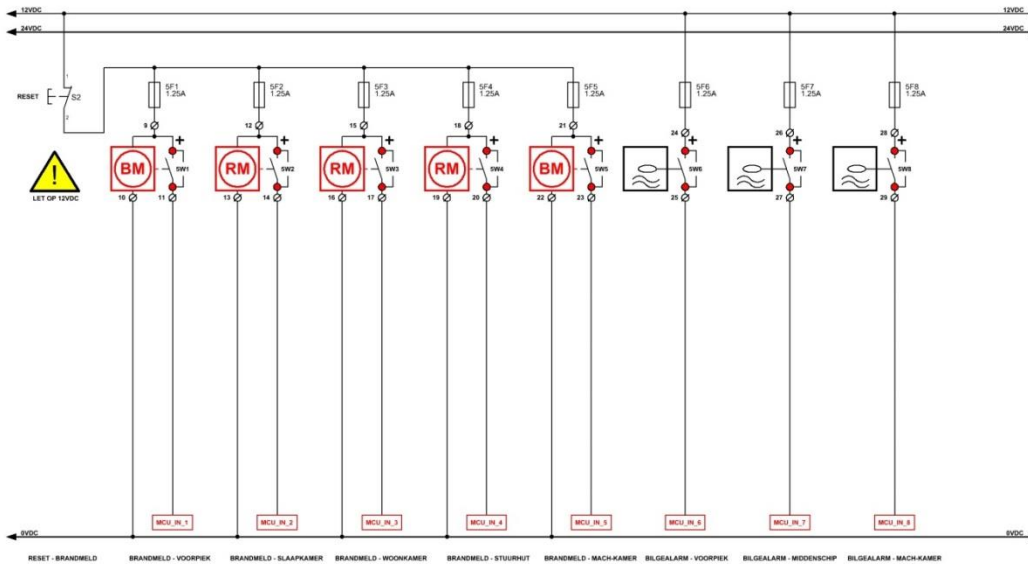
Op deze manier worden de brandmelders, brandstof en waters tanks, en het bilge water in het schip in de gaten gehouden.

Op blz. 138 en Blz. 139 een paar schema's.

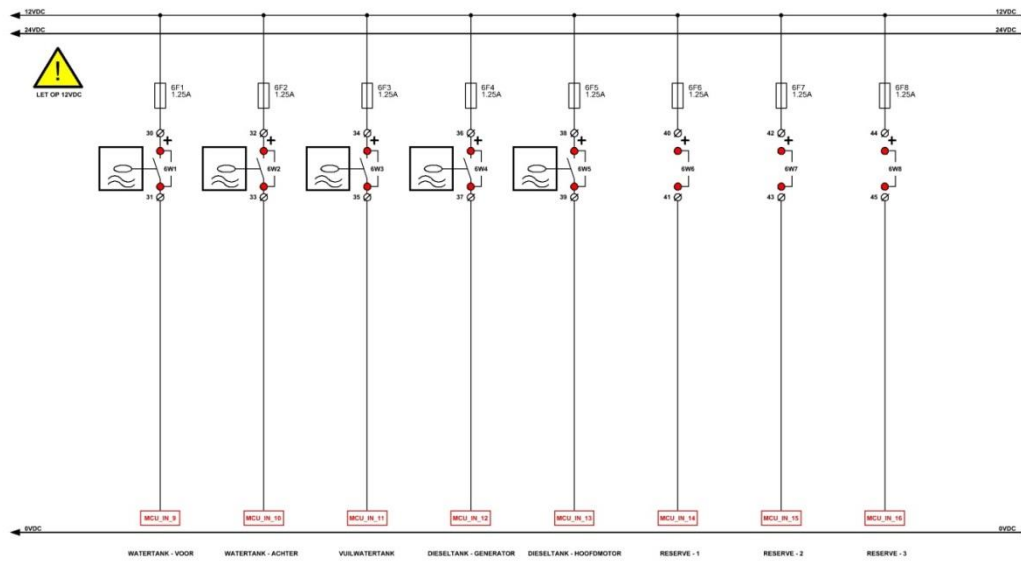
Hieronder het schema van de voeding, pc, interface, enz.



Hieronder het schema van de brandmelders, en de vlotters van de bilge alarmen.



Hier onder het schema van de vlotters van de watertanks, dieseltanks, en de reserve ingangen.



Als pc wordt er gebruik gemaakt van een PC2 computer, dat zijn mooie compacte pc's.



Er zitten zes USB poorten op, een kaart lezer, Lan-1, W-lan, Audio-in en Audio-uit, een DVI aansluiting voor het beeldscherm en als opslag een SSD schijf.

Op de pc draait Windows XP en Profilab.

Het hele systeem draait op 12VDC.

Voor de data overdracht is weer de seriële interface gebruikt die we al eerder in het boek besproken hebben, en een USB serieel converter van Sabrent.

De alarm signalen komen binnen op een interfaceboard waar ook een 16F887 controller op zit. De controller verwerkt de in gekomen data, en verstuurd dit naar Profilab. In de controller zit ook weer een programma dat geschreven is in basic.

Hieronder het programma.

```

'* Name      : BOEK SERIEEL VERZENDEN ALARMEN ZEEMEEUW.
'* Author    : H.van Zwieten.
'* Notice    : Copyright (c) 2014 H.v.Z.
'*           : All Rights Reserved
'* Date      : 24-6-2014
'* Version   : 1.0
'* Notes     : PROGRAMMA WERKT SAMEN MET PROFILAB EXPER 4.0.
'* Notes     : DATA WORDT SERIEEL VERZONDEN VIA DE USB POORT.

Device 16F887                               ; Processor type

Xtal 10                                     ; Cristal 10Mhz

Asm                                         ; Config settings
CONFIG_REQ
__CONFIG __CONFIG1, HS_OSC & WDT_OFF & DEBUG_OFF & FCMEN_OFF & LVP_OFF &
IESO_OFF & BOR_OFF & CPD_OFF & CP_OFF & MCLR_OFF & PWRTE_ON
__CONFIG __CONFIG2, WRT_OFF & BOR40V
EndAsm

All_Digital true                           ; Alle poorten digitaal

Declare LCD_RSPin PORTD.2                  ; Reset display poort D.2
Declare LCD_ENPin PORTD.3                  ; Enable display poort D.3
Declare LCD_DTPin PORTD.4                  ; Data display poort D.4 t/m
D.7

Declare Rsout_Pin PORTB.1                  ; Seriele data uit
Declare Rsin_Pin PORTB.0                   ; Seriele data in
Declare Serial_Baud 2400                   ; Baudrate 2400

Symbol BRANDMELD_VOORPIEK                 = PORTA.0   ; Ingang brandmeld voorpiek
Symbol BRANDMELD_SLAAPKAMER               = PORTA.1   ; Ingang brandmeld slaapkamer
Symbol BRANDMELD_WOONKAMER                = PORTA.2   ; Ingang brandmeld woonkamer
Symbol BRANDMELD_STUURHUT                 = PORTA.3   ; Ingang brandmeld stuurhut
Symbol BRANDMELD_MACHINEKAMER            = PORTA.5   ; Ingang brandmeld machinekamer
Symbol BILGEALARM_VOORPIEK                = PORTE.0   ; Ingang bigealarm voorpiek
Symbol BILGEALARM_MIDSCHEEPS             = PORTE.1   ; Ingang bigealarm midscheeps
Symbol BILGEALARM_MACHINEKAMER           = PORTE.2   ; Ingang bigealarm machinekamer
Symbol WATERTANK_VOOR                     = PORTC.3   ; Ingang watertank voor
Symbol WATERTANK_ACHTER                   = PORTD.0   ; Ingang watertank achter
Symbol VUILWATERTANK                      = PORTD.1   ; Ingang vuilwatertank
Symbol DIESELTANK_GENERATOR                = PORTA.4   ; Ingang dieseltank generator
Symbol DIESELTANK_HOOFDMOTOR              = PORTC.7   ; Ingang dieseltank hoofdmotor
Symbol RESERVE_1                          = PORTC.6   ; Ingang reserve - 1
Symbol RESERVE_2                          = PORTC.5   ; Ingang reserve - 2
Symbol RESERVE_3                          = PORTC.4   ; Ingang reserve - 3

```

```

Symbol ZOEMER           = PORTC.0      ; Uitgang alarmen
Symbol HOORN            = PORTC.1      ; Uitgang brandalarm

Clear                   ; Wis geheugen

Cls                     ; Wis display

DelayMS 500             ; Pauze 0.5 sec

;543210                   ; Hulpregel poort A
PORTA = %000000        ; Maak poort A laag
TRISA = %111111        ; Poort_A I/O

;543210                   ; Hulpregel poort B
PORTB = %000000        ; Maak poort B laag
TRISB = %000001        ; Poort_B I/O

;76543210                 ; Hulpregel poort C
PORTC = %00000000      ; Maak poort C laag
TRISC = %11111000      ; Poort_C I/O

;76543210                 ; Hulpregel poort D
PORTD = %00000000      ; Maak poort D laag
TRISD = %00000011      ; Poort_D I/O

;210                       ; Hulpregel poort E
PORTE = %000           ; Maak poort E laag
TRISE = %111           ; Poort_E I/O

;76543210                 ; Hulpregel analoog
ADCON0 = %00000001     ; ADCON0 register analoog

;543210                   ; Hulpregel analoog
ANSELH = %000000      ; ANSEL register analoog

;-----
; PROGRAMMA - UITLEZEN ALARMEN - ZEEMEEUW.
;-----

;-----
; SERIEEL - UIT - BRANDMELDING.
;-----

RUN:
  If BRANDMELD_VOORPIEK = 0 Then
    RSOut 10
    DelayMS 50
    Print At 1,1,"BRAND_ALARM"
    Print At 2,1,"VOORPIEK"
    DelayMS 1750
    HOORN = 1
    DelayMS 250
  EndIf

  HOORN = 0
  Cls

```

```

If BRANDMELD_SLAAPKAMER = 0 Then
  RSOut 11
  DelayMS 50
  Print At 1,1,"BRAND_ALARM"
  Print At 2,1,"SLAAPKAMER"
  DelayMS 1750
  HOORN = 1
  DelayMS 250
EndIf

HOORN = 0
Cls

If BRANDMELD_WOONKAMER = 0 Then
  RSOut 12
  DelayMS 50
  Print At 1,1,"BRAND_ALARM"
  Print At 2,1,"WOONKAMER"
  DelayMS 1750
  HOORN = 1
  DelayMS 250
EndIf

HOORN = 0
Cls

If BRANDMELD_STUURHUT = 0 Then
  RSOut 13
  DelayMS 50
  Print At 1,1,"BRAND_ALARM"
  Print At 2,1,"STUURHUT"
  DelayMS 1750
  HOORN = 1
  DelayMS 250
EndIf

HOORN = 0
Cls

If BRANDMELD_MACHINEKAMER = 0 Then
  RSOut 14
  DelayMS 50
  Print At 1,1,"BRAND_ALARM"
  Print At 2,1,"MACINEKAMER"
  DelayMS 1750
  HOORN = 1
  DelayMS 250
EndIf

HOORN = 0
Cls

```

```

;-----
; SERIEEL - UIT - BILGEALARM.
;-----

```

```

If BILGEALARM_VOORPIEK = 0 Then
  RSOut 15
  DelayMS 50
  Print At 1,1,"BILGE_ALARM"
  Print At 2,1,"VOORPIEK"
  DelayMS 1750
  ZOEMER = 1
  DelayMS 250
EndIf

ZOEMER = 0
Cls

If BILGEALARM_MIDSCHEEPS = 0 Then
  RSOut 16
  DelayMS 50
  Print At 1,1,"BILGE_ALARM"
  Print At 2,1,"MIDSCHEEPS"
  DelayMS 1750
  ZOEMER = 1
  DelayMS 250
EndIf

ZOEMER = 0
Cls

If BILGEALARM_MACHINEKAMER = 0 Then
  RSOut 17
  DelayMS 50
  Print At 1,1,"BILGE_ALARM"
  Print At 2,1,"MACHINEKAMER"
  DelayMS 1750
  ZOEMER = 1
  DelayMS 250
EndIf

ZOEMER = 0
Cls

;-----
; SERIEEL - UIT - TANK - NIVEAU.
;-----

If WATERTANK_VOOR = 0 Then
  RSOut 18
  DelayMS 50
  Print At 1,1,"WATERTANK_VOOR"
  Print At 2,1,"BIJVULLEN"
  DelayMS 1750
  ZOEMER = 1
  DelayMS 250
EndIf

ZOEMER = 0
Cls

```

```

If WATERTANK_ACHTER = 0 Then
  RSOut 19
  DelayMS 50
  Print At 1,1,"WATERTANK_ACHTER"
  Print At 2,1,"BIJVVULLEN"
  DelayMS 1750
  ZOEMER = 1
  DelayMS 250
EndIf

```

```

ZOEMER = 0
Cls

```

```

If VUILWATERTANK = 0 Then
  RSOut 20
  DelayMS 50
  Print At 1,1,"VUILWATERTANK"
  Print At 2,1,"LEGEN"
  DelayMS 1750
  ZOEMER = 1
  DelayMS 250
EndIf

```

```

ZOEMER = 0
Cls

```

```

If DIESELTANK_GENERATOR = 0 Then
  RSOut 21
  DelayMS 50
  Print At 1,1,"DIESEL_GENERATOR"
  Print At 2,1,"BIJVVULLEN"
  DelayMS 1750
  ZOEMER = 1
  DelayMS 250
EndIf

```

```

ZOEMER = 0
Cls

```

```

If DIESELTANK_HOOFDMOTOR = 0 Then
  RSOut 22
  DelayMS 50
  Print At 1,1,"DIESEL_HOOFD_MOT"
  Print At 2,1,"BIJVVULLEN"
  DelayMS 1750
  ZOEMER = 1
  DelayMS 250
EndIf

```

```

ZOEMER = 0
Cls

```

```

;-----
; SERIEEL - UIT - *****.
;-----

```



```

If RESERVE_1 = 0 Then
  RSOut 23
  DelayMS 50
  Print At 1,1,"RESERVE_1"
  Print At 2,1,"*****"
  DelayMS 1750
  ZOEMER = 1
  DelayMS 250
EndIf

ZOEMER = 0
Cls

```

```

If RESERVE_2 = 0 Then
  RSOut 24
  DelayMS 50
  Print At 1,1,"RESERVE_2"
  Print At 2,1,"*****"
  DelayMS 1750
  ZOEMER = 1
  DelayMS 250
EndIf

ZOEMER = 0
Cls

```

```

If RESERVE_3 = 0 Then
  RSOut 25
  DelayMS 50
  Print At 1,1,"RESERVE_3"
  Print At 2,1,"*****"
  DelayMS 1750
  ZOEMER = 1
  DelayMS 250
EndIf

ZOEMER = 0
Cls

```

```

;-----
; SERIEEL - UIT - CONTROLE - FUNCTIE.
;-----

```

```

  RSOut 1
  DelayMS 100
  RSOut 0
  DelayMS 1000
GoTo RUN

End

```

Zoals je kan zien in het programma wordt er een waarde serieel verstuurd als er een ingang laag wordt, de inkomende signalen van de rookmelders en vlotters zijn positief en worden op de interface print omgezet naar een negatief signaal. Het inkomende signaal wordt dus geïnverteerd, als er dus een nul op de controller verschijnt voort deze een actie uit.

In Profilab wordt de inkomende data vergeleken, en als die afwijkt zal er een lamp gaan branden bij het desbetreffende alarm.

Tot zover het basic programma, we gaan nu de frontplaat plaatsen.

Hieronder de frontplaat.

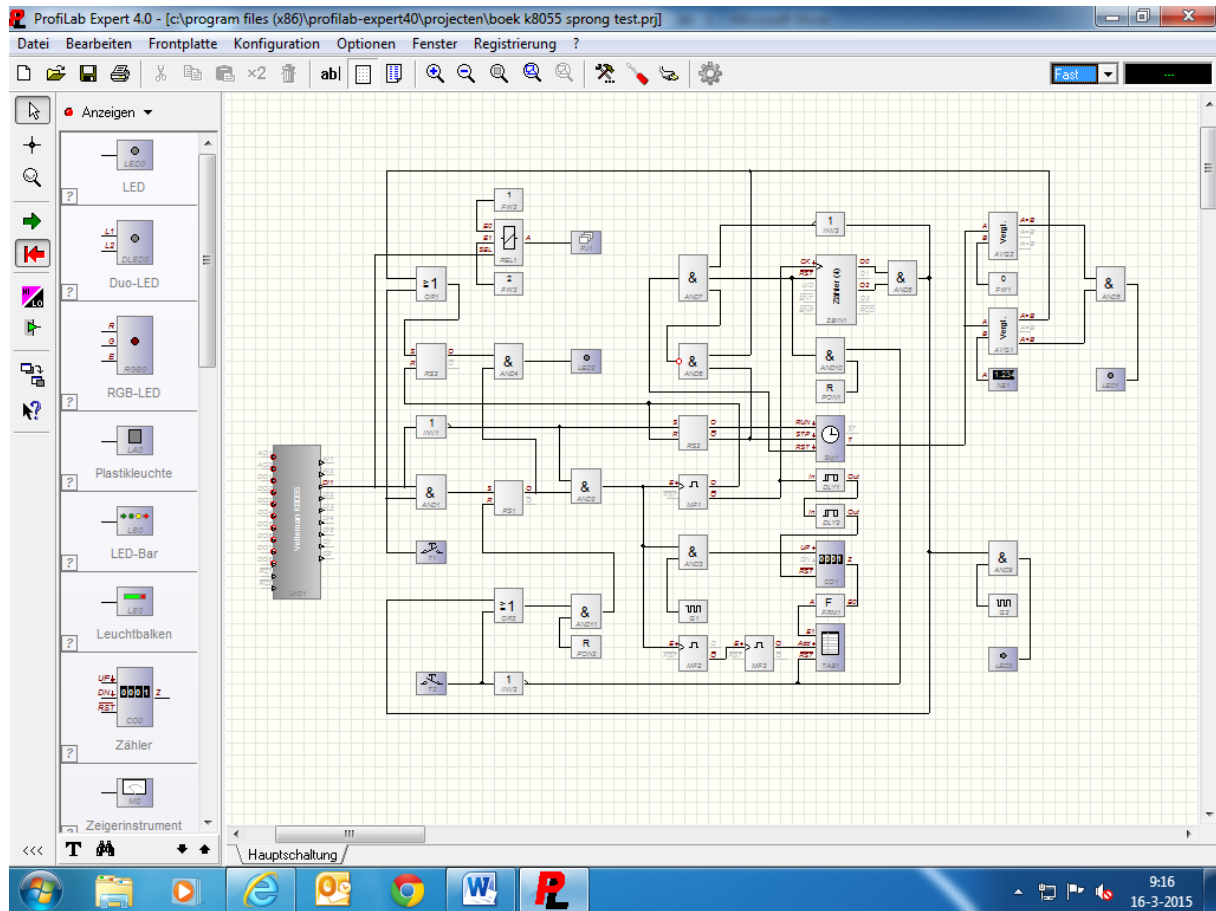


Op de frontplaat zijn alle alarmen te zien, voor ieder alarm staat een lamp die gaat knipperen zodra er iets aan de hand is. Met de reset knop kan het alarm gereset worden, maar dat kan niet eerder dan dat het probleem verholpen moet zijn. Als er bijv. een bilge alarm is in de machinekamer, dan zal dat eerst verholpen moeten worden. Zolang daar water staat kan het alarm niet gereset worden, dat alarm zal dus blijven gaan. Ook zie je nog een lampje voor RX – DATA – SIGNAAL staan, dat knippert +/- iedere seconden zolang er maar een serieel signaal aanwezig is. Als er serieel iets niet in orde is knippert de lamp niet meer, er is dus een mate van controle op het signaal.

Tot zover over het voorbeeld.

8.3: Sprong test training

Dit is een voorbeeld van een spring training voor diverse sporten.

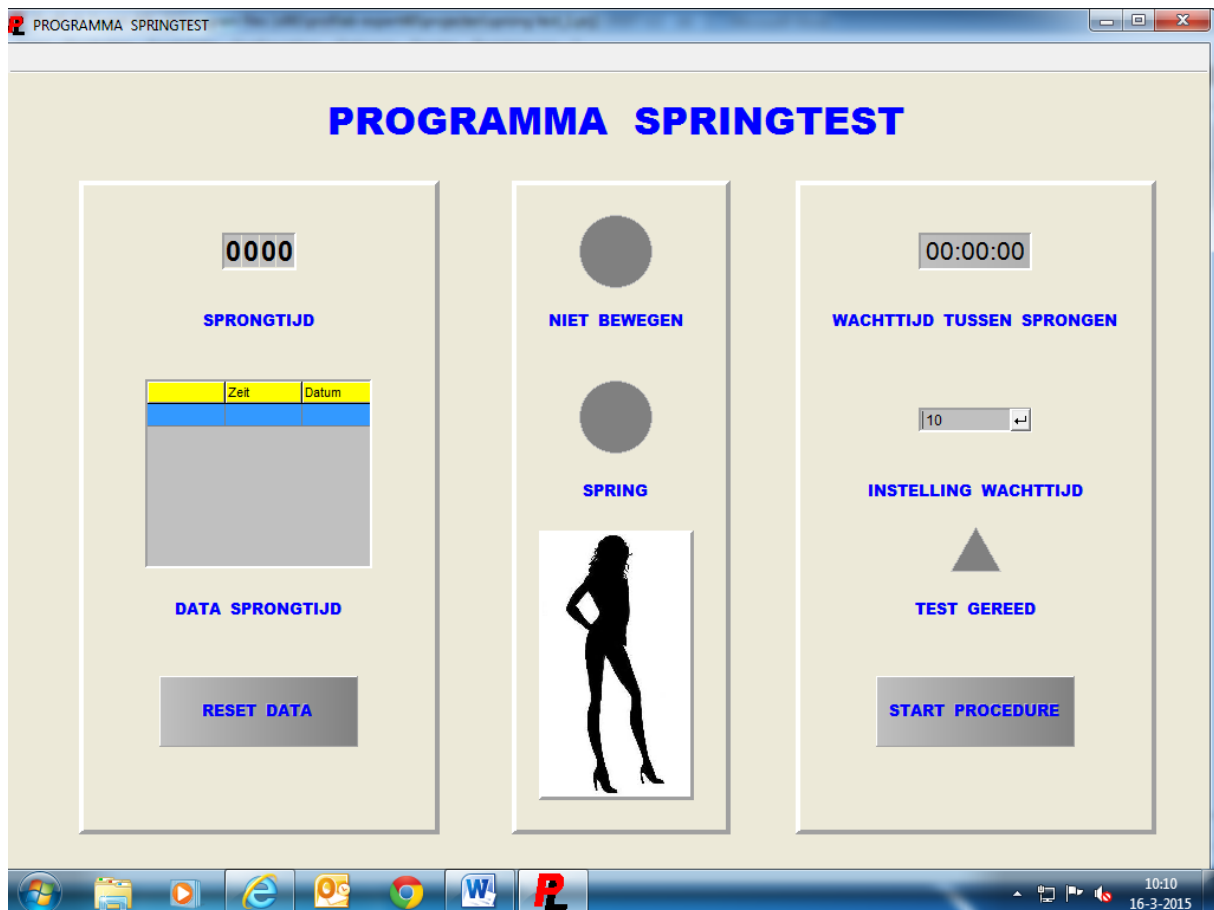


Hier wordt gebruikgemaakt van de K8055 module om de data binnen te halen. Op de ingang van de K8055 zit een contactmat zodra er iemand op de mat staat is het contact gemaakt, en als de persoon springt is het contact verbroken. Op deze manier kan je zien hoelang iemand in de lucht is geweest.

Voor de rest zien we een aantal and poorten, or poorten, inverters, flipflops, delay modules, vergelijker modules, een teller module, een paar leds, een display module, een analoog relais module, en een paar drukknoppen.

Er is gebruik gemaakt van De K8055, maar dat had ook de seriële poort of de usb poort van de pc kunnen zijn. Wel met een daarvoor ontworpen interface natuurlijk.

Op Blz. 148 staat de frontplaat.



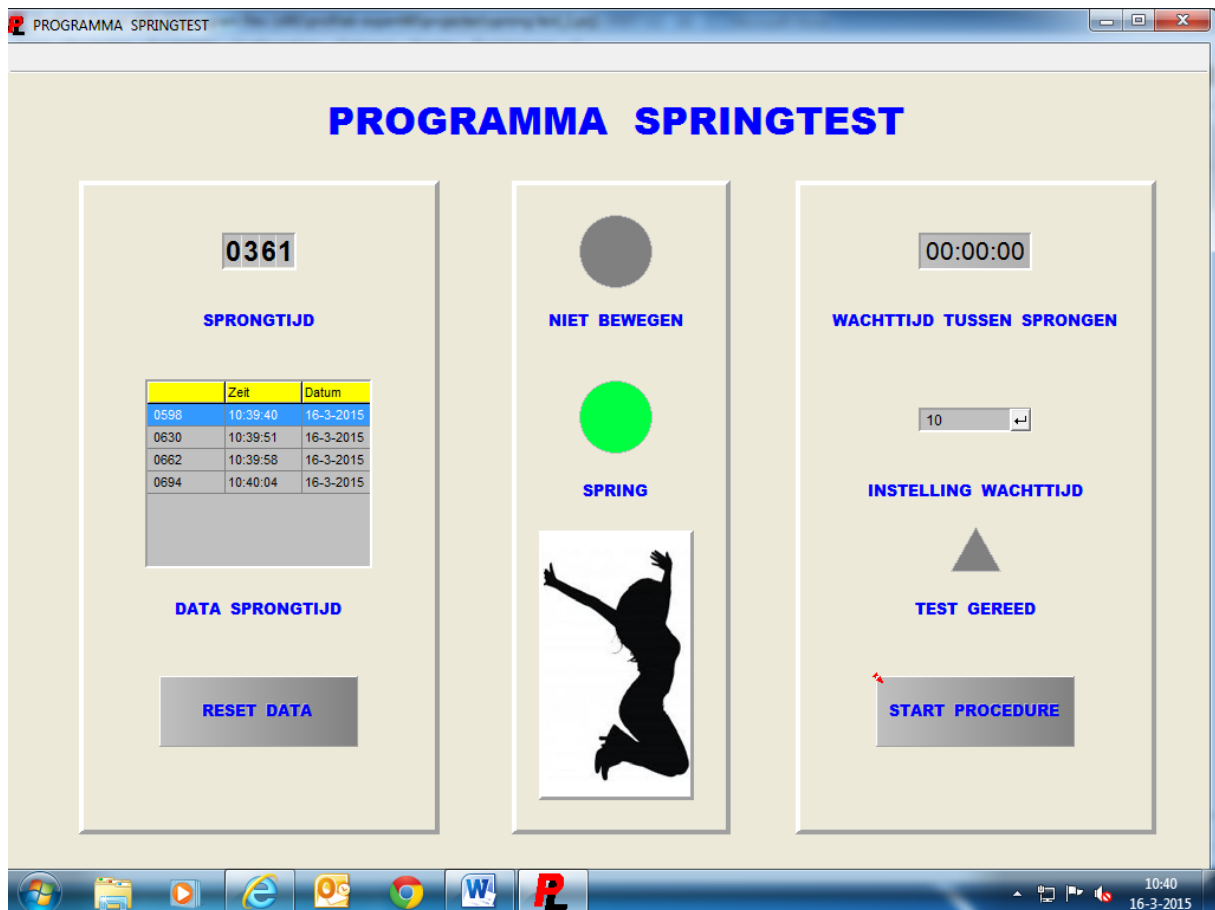
Dit is de frontplaat voor de spring test.

Zoals je kan zien staan er twee drukknoppen op de frontplaat, één om het programma te starten, en één om de data te resetten. Je ziet boven instelling wachttijd een veld staan waar je de wachttijd in seconden in kan geven. Boven aan zie je een timer staan deze gaat lopen als er gesprongen is, als het contact met de mat weer gemaakt is gaat ook de rode lamp aan. Als de ingestelde wachttijd verstreken is dan spring de groene lamp aan.

De tijd dat men in de lucht is, is de sprong tijd. De data van de sprongen wordt in een tabel gezet, zodat deze opgeslagen kan worden om bijv. later te vergelijken. Op deze manier kan men zien of de persoon in kwestie vooruit gaat of gelijk blijft met zijn of haar sprongen.

De data kan ook weer in een Word of Excel bestand gezet worden als men zou willen. Daar is hier geen knop voor aanwezig op de frontplaat, omdat men dat niet nodig vond. De data wordt opgeslagen door op de tabel te gaan staan, en rechts te klikken. Daar kan men dan kiezen hoe het opgeslagen moet worden.

Op Blz. 149 nog een voorbeeld van de frontplaat, maar nu met data er op.



Hier het programma in actie. Op de frontplaat zie je nu dat de persoon net aan het springen is, je ziet ook dat de lamp op groen staat. Er is ook al wat data verzameld van de sprongen, zoals te zien is in de tabel.

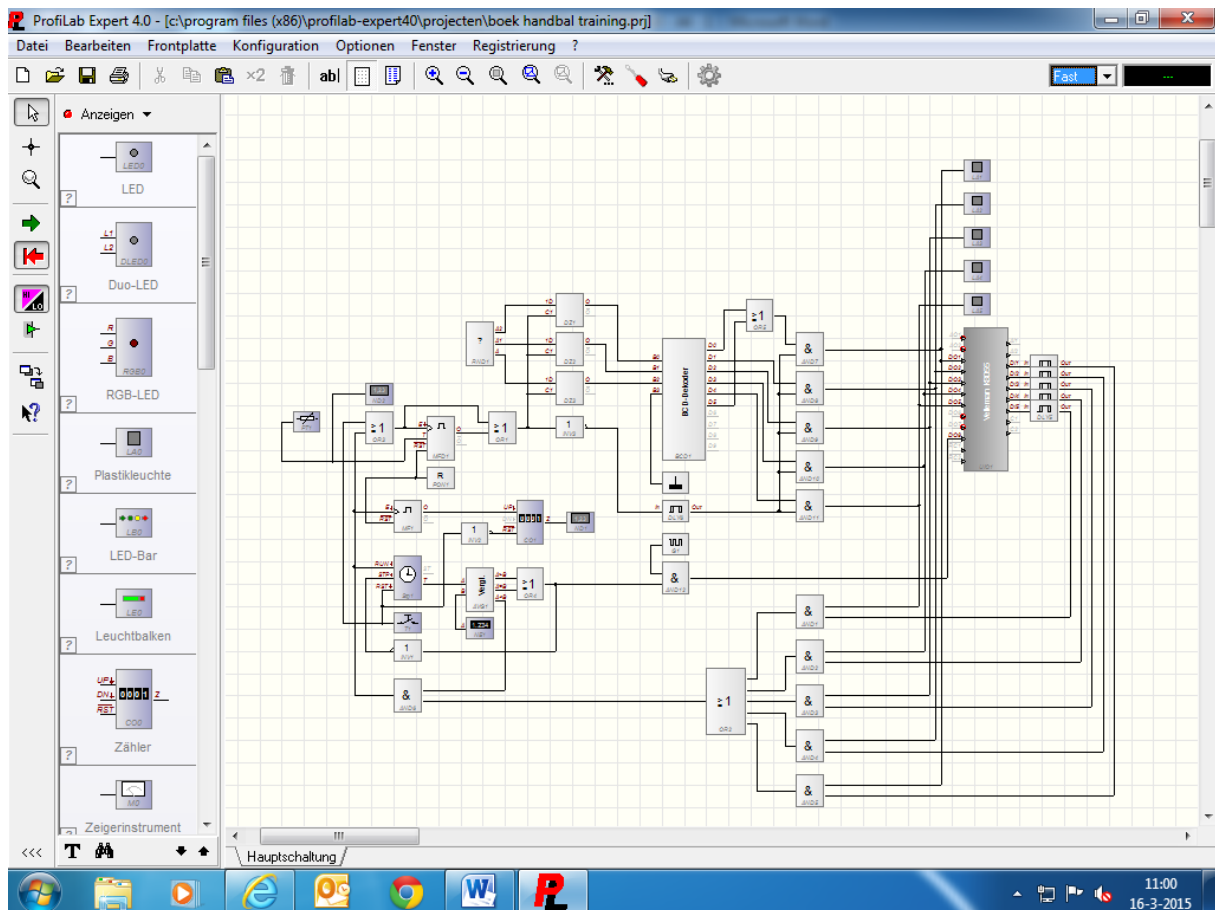
Op deze manier kan je dus een data bank opbouwen van de sprongen die gemaakt zijn. Tot zover over het sprong test programma.

8.4: Handbal training

Dit is een programma voor handbal training, en wel het doel schieten. Alleen wordt er geen bal gebruikt. Dit programma is ooit voor handbal gemaakt, maar het kan ook gebruikt worden voor andere sporten waar een doel bij gebruikt wordt.

De bedoeling van het programma is om de reactie snelheid van de doelman te verbeteren, zodat hij bij een echte wedstrijd beter reageert op de bal.

Op Blz. 150 het programma.



Dit is het programma voor de handbal training.

Bij dit programma wordt er ook gebruik gemaakt van de K8055. Er worden vijf ingangen en vijf uitgangen gebruikt van de interface kaart. Er had natuurlijk ook weer gebruik gemaakt kunnen worden van een microcontroller, maar dat wilde men niet.

Voor de rest is er gebruik gemaakt van: or poorten, and poorten, flipflops, vergelijk modules, displays, een ingave module, een BCD encoder, een potmeter, een drukknop, een puls gever, een paar delay modules, en wat lampen.

Alles is weer met elkaar verbonden zodat we een werkend programma krijgen.

De lampen gaan in willekeurige volgorde aan, als er een lamp brand moet deze zo snel mogelijk aan getikt worden in een bepaalde tijd. Dat levert dan een X aantal punten op.

De frontplaat heb ik weer als volgt gemaakt.

Zie Blz. 151.



Hier de frontplaat voor de handbal training.

Zoals je kan zie op de frontplaat kan je in het onderste veld de duur van de training instellen. Dit is weer de tijd in seconden. Met de potmeter stel je de tijd in dat de lampen aan zijn, hoe korter de tijd is des te sneller zullen de lampen wisselen. Boven de potmeter zie je de ingestelde tijd op het display staan. De gescoorde punten binnen de ingestelde tijd staan op het display ernaast.

De lampen verspringen in een willekeurige volgorde, dus je kan van tevoren niet inschatten welke lamp dat gaat worden. Op deze manier blijft men geconcentreerd en kan men de reactie snelheid van de doelman trainen.

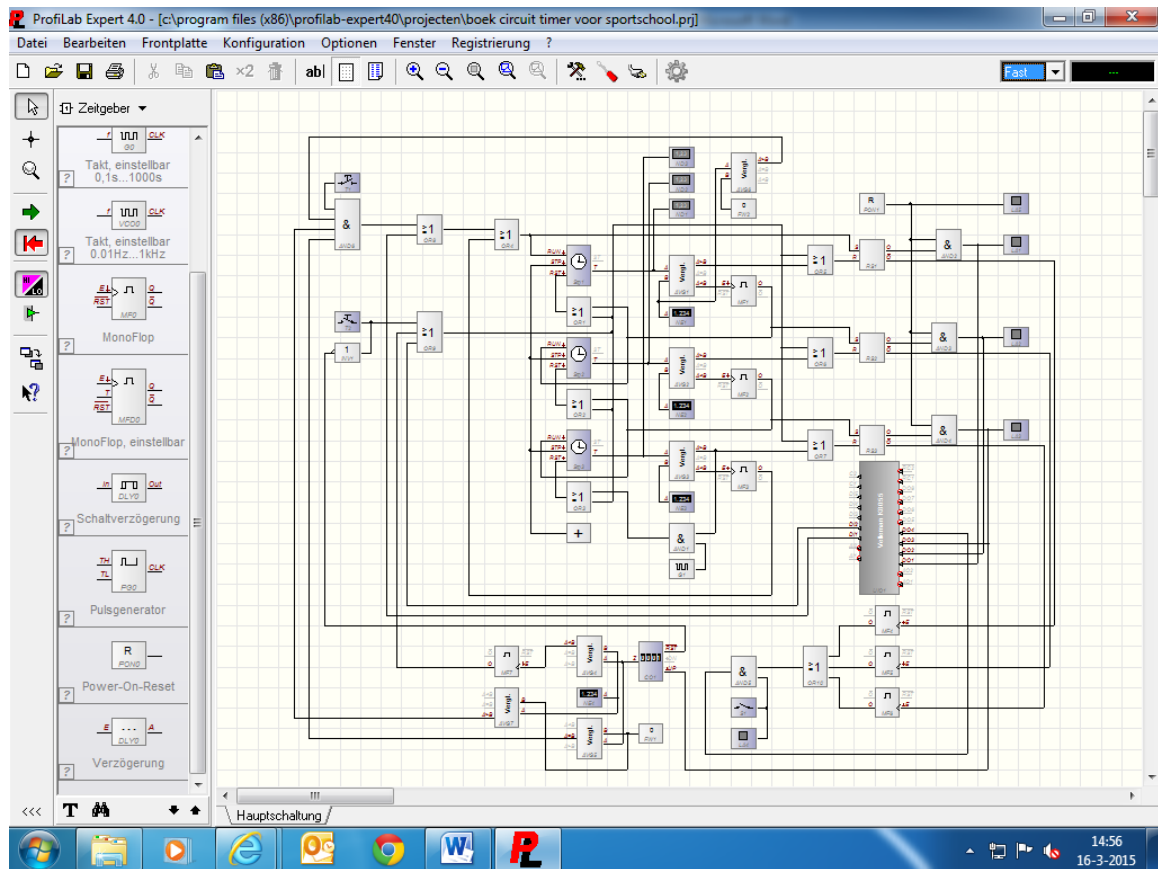
Op of naast de lamp bevinden zich de schakelaars die ingedrukt moeten worden als de lamp brand. Op deze manier heeft men dus geen bal nodig tijdens de training.

Er wordt hier gebruik gemaakt van maar vijf lampen, dat komt door de beperking van de K8055. Die heeft maar vijf ingangen, maar dit is wat men wilde.

Tot zover het programma voor de handbal training.

8.5: circuit training

Dit is een programma voor in de sportschool, circuit training genaamd.



Ook hier is weer de K8055 gebruikt als interface.

De rest van de modules zijn: or poorten, and poorten, klok modules, vergelijk modules, ingave modules, drukknoppen, displays, flipflops, monoflop modules, een puls gever, een reset module, en lampen.

Dit alles is weer met elkaar verbonden zoals je op het werkblad kan zien. De K8055 stuurt een stoplicht aan. De kleuren van het stoplicht zijn groen, geel, en rood. De bedoeling van het programma is dat de gebruiker van het circuit voor een bepaalde tijd een onderdeel afhandelt. Als de groene lamp brand dan moet het onderdeel (bijv. roeien) met de hoogst mogelijke inspanning in een bepaalde tijd afgehandeld worden. Als de gele lamp brand moet er rustig uit geroeid worden voor een bepaalde tijd, en als de rode lamp brand mag men even uitrusten voor een bepaalde tijd.

Een circuit bestaat uit een aantal apparaten, vaak zijn dat er negen. Maar het kan ook meer of minder zijn.

Op Blz. 153 staat de frontplaat.



Hier boven de frontplaat van het circuit.

Zoals je kan zien staan er vier velden waar je data in kan voeren, in het groene veld geef je de tijd in voor de inspanning van bijv. roeien. Dat wordt dan bijv. 50 seconden gedaan. Op het gele veld geef je de tijd in voor het uit roeien, (rustig aan dus) bijv. 25 seconden. En in het rode veld geef je de tijd aan dat je uit kan rusten en naar het volgende apparaat gaat. Als de rode lamp uit is springt de groene lamp weer aan, en kan men de volgende oefening starten op een ander apparaat.

In het veld (aantal apparaten) geef je het aantal apparaten aan die je wil gebruiken. Stel dat er negen apparaten zijn en je wilt één ronde doen dan geef je dus negen in. Wil je twee rondes doen dan geef je achttien in. Als je bijv. negen apparaten gehad hebt dan wordt het programma gereset, het programma kan ook zelf gereset worden met de drukknop op het display.

De teller op de frontplaat houdt de stand bij.

Er zit ook een zoemer op het systeem, die kan je in of uitschakelen op de frontplaat. De zoemer geeft aan als er een lamp van de ene naar de andere kleur springt.

Tot zover het programma voor de fitness training.

Op Blz. 154 staat een voorbeeld van zo'n circuit.



Zoals men kan zien staan hier negen verschillende apparaten die gebruikt worden voor circuit training. Het kan ook een combinatie zijn van andere apparaten die hier getoond worden, maar dat ligt aan de sportschool houder.

Dit was zomaar een greep uit de praktijk voorbeelden, deze voorbeelden worden dagelijks gebruikt.

Hiermee zijn we ook aan het einde van het boek gekomen, ik hoop dat er genoeg inspiratie uit de voorbeelden gehaald kan worden om er zelf ook leuke projecten mee te maken. Zoals je gezien hebt kan je heel veel doen met Profab expert 4.0, maar ook in combinatie met een controller zoals in het boek getoond en besproken is.