Toegepaste Informatica / Elektronica-ICT

Herstelsysteem ramen

Onderdeel van de stage   
ondersteund door de

AP Hogeschool

en uitgevoerd op en begeleid door het bedrijf

Nationale Maatschappij der Belgische Spoorwegen

Jelte Boumans

Specialisatie IT & Internet of Things – Minor Maker

# Inhoud

[Inhoud 1](#_Toc135583210)

[Voorwoord 2](#_Toc135583211)

[Inleiding 3](#_Toc135583212)

[Samenvatting 4](#_Toc135583213)

[Afkortingen 5](#_Toc135583214)

[1 Hardware 6](#_Toc135583215)

[1.1 Versie 1 6](#_Toc135583216)

[1.1.1 Blokdiagram 6](#_Toc135583217)

[1.1.2 Keuze van componenten 7](#_Toc135583218)

[1.1.3 Bill of Materials 9](#_Toc135583219)

[1.1.4 Schema 10](#_Toc135583220)

[1.1.5 PCB 12](#_Toc135583221)

[1.1.6 Fouten 14](#_Toc135583222)

[1.1.7 Conclusie 16](#_Toc135583223)

[1.2 Versie 2 17](#_Toc135583224)

[1.2.1 Blokdiagram 17](#_Toc135583225)

[1.2.2 Keuze van componenten 17](#_Toc135583226)

[1.2.3 Bill of Materials 17](#_Toc135583227)

[1.2.4 Schema 17](#_Toc135583228)

[1.2.5 PCB 17](#_Toc135583229)

[1.2.6 Conclusie 17](#_Toc135583230)

# Voorwoord

Van NMBS en AP heb ik de opdracht gekregen: een nieuw herstelsysteem ontwikkelen voor het vervangen van de ramen in de MR08 Desiro treinen. Deze opdracht was voor mij een hele uitdaging zowel op vlak van software ontwikkeling als op vlak van hardware design.

Gelukkig kon ik reken op steun van meerdere collega’s van NMBS en van AP lectoren.

Mijn bijzondere dank gaat dan ook uit naar: Hansjörg Van Rompay voor zijn eindeloos geduld en feedback op mijn project. Tibo Suys en Aaron ... voor hun hulp en feedback bij het designen van de hardware. Tom ... en Arno ... voor hun hulp bij de ontwikkeling en integratie van de software. Tenslotte Geert Vanhulle, mijn stagebegeleider voor zijn niet aflatende steun.

# Inleiding

Het onderwep van mijn project was een nieuw herstelsysteem ontwikkelen dat de technici kunnen gebruiken wanneer ze de beschadigde ramen moeten vervangen van de MR08 Desiro trein.

Nu wanneer de technici de ramen vervangen moeten er een aantal parameters genoteerd worden. Bijvoorbeeld: temperatuur, luchtvochtigheid en openingsdatum van bepaalde gebruikte producten. Deze parameters worden opgeslagen door ze te noteren in een Excel document, wat niet zeer handig/praktisch is.

Het doel van mijn project is: de werkmethode om de nodige parameters op te slaan, te integreren in het systeem van NMBS, genaamd BeeTree. Ik moet een interface ontwikkelen die compatibel is met BeeTree, zodat de teamlead een herstelling kan aanmaken, de schouwer de ramen kan selecteren die nood hebben aan vervanging en de technici bij elke raam de nodige parameters kan invullen. Tenslotte moet ik ook het noteren van de temperatuur en luchtvochtigheid gemakkelijker maken door een datalogger te ontwerpen die deze parameters zal meten en automatisch invullen in de interface. Dit zal mogelijk zijn door deze te verbinden met de tablet van de technici via USB-C.

# Samenvatting

Mijn project heb ik kunnen verdelen in 3 verschillende onderdelen:

* Hardware
* Firmware
* Software

Het hardware gedeelde bestaat uit het ontwerpen van de datalogger. Dat houd in: een PCB ontwerpen met de nodige componenten om de datalogger te laten werken. Deze PCB moet ook klein zijn zodat het gemakkelijk te gebruiken is met een tablet. Er zal ook een behuizing (of case) moeten ontworpen worden voor de PCB te beschermen. Deze zal gerealiseerd worden met een 3D printer.

Het firmware gedeelde bestaat uit het schrijven van de nodige firmware om de microcontroller van de datalogger correct te laten werken. Deze firmware zal geschreven zijn in Arduino.

Het software gedeelde bestaat uit het ontwerpen van een interface dat de techici zullen gebruiken. Eens een goed ontwerp is gemaakt voor de interface zal deze worden gerealiseerd door gebruik te maken van het PHP framework genaamd Smarty. Dit is het framework waarin BeeTree geschreven is.

Deze documentatie zal gestrucureerd zijn rond deze 3 delen van het project.

# Afkortingen

| Term | Omschrijving |
| --- | --- |
| PCB | Printed Circuit Board - Printplaat |
| PHP | Personal Home Page / PHP: Hypertext Preprocessor |
| USB | Universal Serial Bus |
| PPTC | Polymeric Positive Temperature Coefficient |
| UART | Universal Asynchronous Receiver-Transmitter |
| IC | Integrated Circuit |
| I2C | Inter-Integrated Circuit |
| I/O | Input/Output |
| OLED | Organic Light-Emitting Diode |
| DTR | Data Terminal Ready |
| ADC | Analog to Digital Converter |

# Hardware

De hardware bestaat uit de datalogger en zijn behuizing. Er zijn in totaal 2 versies gemaakt van de datalogger. In dit hoofdstuk zal ik voor elke versie de hardware analyse en mijn redenering uitleggen.

## Versie 1

### Blokdiagram

A picture containing text, screenshot, diagram, font

Description automatically generated

Figuur 1 blokdiagram versie 1

In deze diagram word elke component van de datalogger voorgesteld als een blok. Deze blokken zijn aan elkaar verbonden met pijlen die de vloei van de data aantoond. Dus de pijl tussen de “Atmega328p” en “16MHz crystal” betekend dat de data van het “16MHz crystal” naar de “Atmega328p” vloeit. Op elke pijl staat ook genoteerd wat voor data het is. Bij hetzelfde voorbeeld als daarnet vermeld is het een kloksignaal dat de “Atmega328p” zal ontvangen.

### Keuze van componenten

* USB-C Connector
  + De PCB moet en manier om van een bron spanning en stroom te ontvangen en te kunnen communiceren met de tablets. USB is hier de logische oplossing voor want het kan bijde behoeften vervullen. Aangezien de tablets USB-C gebruiken is de logische keuze dus een mannelijke USB-C connector.
* Polyfuse
  + Ik vind dat het nodig is om de PCB en tablet te beschermen indien er een kortsluiting is. Hiervoor is een polyfuse (of een betere term hiervoor een PPTC) zeer handig, het zal de stroom blokkeren indien er een kortsluiting. Na een tijdje zal deze terug beginnen geleiden, dus deze moet niet vervangen worden zoals een smeltzekering. Ik koos voor een PPTC met een “hold current” van 500mA, dit is zeker meer dan de schakeling onder normale omstandigheden nodig zal hebben.
* USB to UART bridge
  + De tablet/computer moet natuurlijk kunnen communiceren met de microcontroller van de datalogger. Dit gaat niet zo maar doordat USB en de microcontroller niet hetzelfde protocol gebruiken (de microcontroller gebruikt UART). Deze IC zal het USB protocol vertalen naar UART en vice versa. De reden dat ik voor de FT232RL koos is omdat deze ook gebruikt wordt in de Arduino Nano, dus ik ben al bekend met deze IC.
* Atmega328p
  + Deze microcontroller is ook dezelfde als op de Arduino Nano. Ik koos deze voor verschillende redenen: de behuizing is zeer klein, niet al te veel I/O pins want ik heb toch alleen de I2C pinnen nodig, zeer goedkoop en ik bezit al over veel ervaring met deze specifieke microcontroller.
* 16MHz crystal
  + Het word aangeraden om een extern kristal te gebruiken, dit zal zorgen voor een accurater kloksignaal en een snellere executie van de firmware. Het is ook nodig wanneer de Atmega328p nog geen bootloader heeft, wanneer deze nog nieuw is zonder bootloader kan ik geen gebruik maken van de interne oscillator om deze als 8MHz klok te gebruiken.
* Temp/humidity/... sensor
  + De sensor is het meest belangrijke component van de datalogger, dus hier moet goed over nagedacht worden. Ik koos voor de BME680 van Bosch. Mijn reden hiervoor is dat deze accuraat temperatuur, luchtvochtigheid, luchtdruk en enkele speciale gassen in de lucht. Dit kan deze sensor allemaal in een behuizing van 3mm bij 3mm. Hierdoor is deze ideaal voor mijn datalogger.
* Display
  + Één van de eisen voor de datalogger was de aanwezigheid van een soort scherm. Ik koos voor een OLED scherm voor goede redenen. Een kleine en plat OLED schermpje is veel meer voorkomend dan bijvoorbeeld een klein plat LCD scherm. Een OLED scherm verbruikt ook veel minder stroom vergeleken met een LCD scherm. Dus in de meeste aspecten is OLED simpelweg beter.
* I2C level shifter
  + De sensor en display worden aangestuurd via I2C. Een probleem is dat de master (Atmega328p) een spanning van 5V gebruikt en de slaves (sensor en display) een spanning van 3.3V gebruiken. Ookal is het mogelijk om dit op te lossen zonder hier een IC voor te gebruiken, is dit niet aangeraden en niet veilig. Hierdoor koos ik voor een “level shifter” te gebruiken. Deze component zal de 5V signalen omvormen naar 3.3V signalen en vice versa.

### Bill of Materials

Voor mijn eerste ontwerp van de datalogger heb ik alle componenten bij de leverancier Mouser besteld. Deze leverancies de grootste keuze aan componenten en was de enige waar ik bepaalde componenten kon vinden (zoals het OLED scherm).

Table

Description automatically generated

Figuur 2 Bill of Materials versie 1

### Schema

Diagram, schematic

Description automatically generated

Figuur 3 schema versie 1

Het schema is verdeeld in verschillende onderdelen: USB-C plug, USB to UART bridge, Main MCU, I2C level shifter, external oscillator, Temp/Humidity sensor en OLED display connector met ook nog kleinere onderdelen zoals de Power LED, Polyfuse, Comm LEDs en Reset button. Deze onderdelen zijn eigelijk de blokken van de blokdiagram, maar dan gerealiseerd in vorm van een elektrisch schema.

Elke component is ook in bezit van een aantal passieve componenten zoals weerstanden en condensatoren. Ik zal nog de keuze achter enkele belangrijke passieve componenten uitleggen:

* C1, C2, C3, C5, C6, C7, C12 en C13 dienen allemaal om de storing en pieken te vermijden in de bronspanning. C5 dient eerder voor storing te vermijden bij de ADC referentie spanning. Omdat ik de ADC van de Atmega328p niet gebruik had ik deze condensator eigelijk kunnen weglaten.
* R1, R6 en R7 zijn stroombeperkende weerstanden voor de LEDs.
* R2 en R3 zijn zeer belangrijk voor de werking van de USB-C connector. Doordat deze aangesloten zijn als pull-down weerstanden zal de USB controller in de tablet mijn PCB herkennen als een “current sink” (dit betekent dat het stroom nodig heeft en geen stroom zal leveren). Als deze afwezig zijn zal de PCB geen stroom/spanning ontvangen. Als ze als pull-up weerstanden worden aangesloten zal de USB controller mijn PCB herkennen als een bron van stroom.
* C4 is ook een zeer belangrijke component voor het programmeren van de Atmega328p. Wanneer de FT232RL klaar is om de Atmega328p te programmeren zal de DTR pin laag worden gemaakt, C4 zal dan kort ontladen om de Atmega328p te resetten. Dit heet “auto-reset” en zonder deze verbinding zou de gebruiker elke keer op het juiste moment op de reset knop moeten drukken om te Atmega328p te programmeren.
* R8, R9, R10, R11, R12, R13, R4 zijn allemaal pull-up weerstanden. R15 is een pull-down weerstand, de reden dat deze zo groot is is omdat in de datasheet vermeldt wordt dat er niet meer dan 12.5µA door deze pin mag.
* C8 en C9 zijn “load capacitors” voor het externe kristal. Het nut van deze condensatoren is om de frequentie van het kloksignaal zo stabiel mogelijk te houden. De Atmega328p kan niet goed functioneren met een onstabiel kloksignaal.
* C10 en C11 zijn “charge pump capacitors”, deze zijn nodig voor de werking van de interne spanningsversterker in het OLED scherm.
* C14 en C15 zijn niet heel duidelijk, in de datasheet wordt er vermeldt dat de VCOMH en VCC pinnen alle 2 deze condensatoren nodig hebben. De reden hiervoor volgens de datasheet is voor “stabilisatie”.

### PCB

A computer screen shot of a circuit board

Description automatically generated with low confidence

Figuur 4 bovenkant PCB versie 1

A picture containing electronics, circuit, electronic engineering, electronic component

Description automatically generated

Figuur 5 onderkant PCB versie 1

A close-up of a computer chip

Description automatically generated with low confidence

Figuur 6 3D render bovenkant PCB versie 1

A close-up of a computer chip

Description automatically generated with low confidence

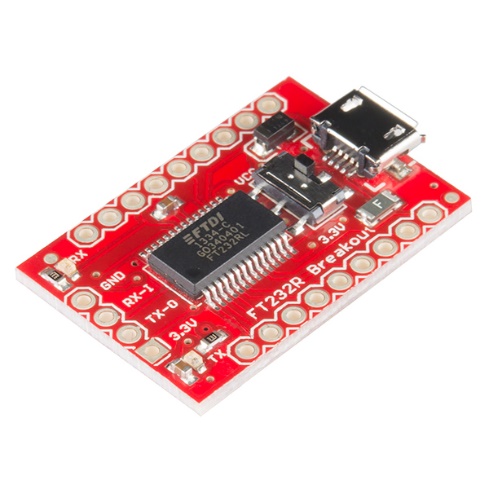
Figuur 7 3D render onderkant PCB versie 1

In deze render is het niet mogelijk, maar natuurlijk in het echt wordt de flexiebele kabel van het OLED scherm via de voorziende gleuf naar de achterkant van de PCB geleid. Deze kan dan gesoldeerd worden aan de voorziene pads.

### Fouten

Meestal bij het ontwerpen van een PCB is niet alles perfect op de eerste poging. Dit was ook zo bij het ontwerpen van de datalogger. Ik heb tijdens het testen enkele fouten ontdekt.

* Zoals eerder vermeldt hebben de sensor en het OLED scherm een bronspanning van 3.3V nodig. De FT232RL beschikt over een interne spanningsregelaar die de voedingspanning van 5V zal omvormen naar 3.3V. Het probleem is dat ik deze gebruik om de sensor en OLED van hun voedingspanning te voorzien. Ookal kan volgens de datasheet deze genoeg stroom leveren, wordt hierdoor de FT232RL nogal warm na een minuutje. Dit zou geen probleem zijn, alleen staat de sensor recht naast deze IC, dit heeft invloed op de temperatuur en luchtvochtigheid metingen.
* Er was ook een probleem met de levering van bepaalde componenten, namelijks de I2C level shifter en de FT232RL. Voor de eerste had ik een andere component gekozen dan de originele, maar ik had de footprint op de PCB niet aangepast omdat ik dacht dat de componenten dezelfde behuizing hadden. Toen ik de PCB soldeerde was het toch wel mogelijk om deze er nog op te monteren, ookal is het niet zeer proper. Voor de FT232RL was het probleem groter, deze was nergens meer leverbaar, maar ik had de PCB al laten bestellen. Dit heb ik opgelost door een “breakout board” te kopen met een FT232RL er op gemonteerd, deze er dan af te solderen en plaatsen op mijn eigen PCB.



Figuur 8 breakout board FT232RL

* De manier waarop de USB-C connector is gemonteerd op de PCB is niet ideaal. Omdat deze maar aan 1 kant van de PCB is gemonteerd is het niet moeilijk om deze af te breken als de behuizing niet goed is gesoldeerd in de pads. Dit zorge ervoor dat een paar dagen na veel gebruik tijdens het debuggen van de PCB en firmware de USB connector was afgebroken. Spijtig genoeg heb ik geen foto’s genomen van de PCB met de USB connector nog verbonden. Ik kan ook geen nieuwe connector monteren aangezien er een groot aantal pads afgescheurd zijn samen met de connector. Mijn oplossing hiervoor was een USB kabel rechtsreeks te solderen aan de nodige pinnen. Dit zorgt ervoor dat de datalogger toch nog zal werken.
* De OLED display werkt ook niet. Dit heb ik vrij laat ontdekt in de stage en heb nog geen reden kunnen ondervinden waarom de display niet zou kunnen werken. Wat een mogelijke reden kan zijn is dat de FT232RL zijn interne spanningsomvormer niet genoeg stroom kan leveren voor beide de sensor en display.
* Tenslotte het grootste probleem met de PCB: de datalogger werkt alleen maar op een computer en niet op een tablet/gsm. De reden hiervoor is de library die ik gebruik voor communicatie tussen de datalogger en website (meer info hierover bij het hoofdstuk over de software). Deze library heeft een speciale eis: de microcontroller moet volledige controle hebben over de USB naar UART bridge. Dus het moet een Atmega zijn met een ingebouwde USB naar UART bridge. De beste optie voor deze eis is de Atmega32U4 van de Arduino Leonardo.

### Conclusie

Ik vindt het spijtig dat de datalogger niet beter was gelukt met de eerste poging. Maar ik ben wel tevreden over het feit dat ik heb kunnen laten zien dat mijn concept van de datalogger wel werkt en beter zal werken eens alle fouten eruit zijn gehaalt.

Het is ook zeer spijtig dat deze versie de enige is die ik heb kunnen realiseren. De reden hiervoor is dat het zeer lang duurde vooralleer ik alle nodige componenten had ontvangen. Een ander probleem is dat ik de PCB ging solderen met een “stencil” en “solder paste”. Op de werkplaats van NMBS hebben ze zulk materiaal niet om te gebruiken, dus om dit op te lossen moest ik dit op school doen. Dit zorgde ervoor dat ik alleen op 1 dag van de week kon solderen, dus veel tijdverspilling als het niet zou lukken op 1 dag.

De eerste versie van de PCB was gesoldeerd ongeveer 3 van de 4 maande in mijn stage. Dus ging er niet genoeg tijd zijn om een 2de versie te bestellen en realiseren. Ik heb dan nog wel een 2de versie ontworpen zodat de volgende die aan mijn project verder zal werken gemakkelijk daar aan kan beginnen.

## Versie 2

### Blokdiagram

### Keuze van componenten

### Bill of Materials

### Schema

### PCB

### Conclusie