**Onderzoek naar een Datalogger voor Validatie van Tijdwaarneming**

**1. Inleiding**

Voor dit onderzoek heb ik mij gericht op het ontwerpen en realiseren van een datalogger die geschikt is voor de validatie van de tijdwaarneming van verkeersregelinstallaties. Als leidraad heb ik onderzoeksvragen geformuleerd tijdens het opstellen van het plan van aanpak.

De eerste onderzoeksvraag was:  
"Wat is een datalogger en hoe werkt datalogging?"

Om deze vraag te beantwoorden, heb ik verschillende bronnen geraadpleegd en mijn eigen ervaring meegenomen in de analyse.

**2. Wat is een datalogger en hoe werkt datalogging?**

Een datalogger is een elektronisch apparaat dat gegevens verzamelt en opslaat over een bepaalde periode, vaak met behulp van ingebouwde of externe sensoren. Deze apparaten zijn doorgaans klein en draagbaar en worden gebruikt voor het monitoren van omgevingscondities zoals temperatuur, luchtvochtigheid, schokken, licht en kanteling.

Door gegevens automatisch en continu te registreren, geven dataloggers een nauwkeurig beeld van de omstandigheden waarin goederen of processen zich bevinden, zoals bij transport of industriële toepassingen.

**3. Onderzoeksvragen**

Om te bepalen welke specificaties de nieuwe datalogger moet hebben, heb ik de volgende onderzoeksvragen opgesteld:

Hoofdonderzoeksvraag:

* Welke hard- en software is het meest efficiënt en effectief voor een datalogger die nauwkeurig de tijdwaarneming kan loggen na een inkomend signaal?

Subvragen:

* Welke hardwarecomponenten zijn geschikt voor het detecteren en verwerken van een inkomend signaal met hoge nauwkeurigheid?
* Welke software-architectuur zorgt voor een efficiënte verwerking en opslag van tijdstempels?
* Wat zijn de prestatie-eisen voor een nauwkeurige tijdsvergelijking?
* Welke opslag- en verwerkingsmogelijkheden zijn nodig voor logging?
* Hoe kan de datalogger worden getest en gevalideerd op nauwkeurigheid?

**4. Vergelijking van bestaande dataloggers bij de politie**

Om een geschikte datalogger te ontwikkelen, heb ik onderzocht welke methodes en apparaten momenteel worden gebruikt binnen het team Forensische Opsporing van de politie Zeeland – West-Brabant. Momenteel zijn er twee systemen in gebruik:

1. Data Translation USB Data Acquisition (DAQ) Module (zie afbeelding 1)
2. Movilog BM22 (zie afbeelding 2)

Afbeelding met tekst, elektronica, Elektronische engineering, Elektrische bedrading

Door AI gegenereerde inhoud is mogelijk onjuist. Afbeelding met elektronica, Elektronisch apparaat, Draagbare spelcomputer, Spelcomputer

Door AI gegenereerde inhoud is mogelijk onjuist.

Afbeelding 1:DAQ module Afbeelding 2:Movilog BM22

De DAQ wordt gebruikt voor validatie van verkeersregelinstallaties, terwijl de BM22 wordt ingezet voor het meten van remvertraging van voertuigen, en specifiek voor het loggen van validatieritten met GPS.

Ik heb de specificaties van de beide dataloggers in een tabel gezet, zie hiervoor tabel A dat als bijlage is opgenomen.

**5. Evaluatie en Ontwikkelpunten van de huidige apparatuur**

Uit eigen ervaring heb ik vastgesteld dat beide apparaten zich in de praktijk hebben bewezen, maar ook ontwikkelpunten hebben.

Movilog BM22 - Sterk concept, maar met verbeterpunten:

* Ongunstige montagelocatie → De plaatsing op de (voor)ruit is onpraktisch, vooral omdat ruiten vaak beschadigd raken bij een aanrijding.
* Niet-oplosbare softwarefout → Een structurele softwarefout kon niet worden hersteld, waardoor een arbeidsintensieve herberekening noodzakelijk is.
* Onhandige voedingsaansluiting → De plus- en min-klemmen moeten direct op de accuklemmen worden bevestigd. Een interne batterij of alternatieve 12V-stekker zou gebruiksvriendelijker zijn.

Gezien de sterke punten van de Movilog BM22, is dit apparaat als basis geschikt, mits de aanpassingen van de ontwikkelpunten worden doorgevoerd.

Aangezien het huidige concept grotendeels functioneel en bewezen effectief is, zijn er geen redenen om hiervan af te wijken. Daarnaast ben ik gebonden aan een aantal landelijk vastgestelde normen (FO-norm), die voorschrijven dat validatie moeten worden uitgevoerd met een datalogger die tijdmetingen uitvoert op 10 kHz

**7. Beantwoording onderzoeksvragen**

Ik heb als eerste de Subvragen beantwoord, want de Subvragen helpen bij de opbouw van het volledige antwoord op de hoofdvraag. Ik heb hiermee structuur aangebracht door van klein naar groot te werken. Eerst details (hardware/software), dan grotere lijnen (prestaties/logging) en uiteindelijk de validatie.

**7.1 Hardware keuze**

**Eerste sub onderzoeksvraag:**

*Welke hardwarecomponenten zijn geschikt voor het detecteren en verwerken van een inkomend signaal met hoge nauwkeurigheid?*

Ik heb de alle benodigde hardware componenten in een tabel gezet, met daarbij de eisen en de huidige hardware die is gebruikt voor de BM22 en de DAQ, zie hiervoor tabel B die als bijlage is toegevoegd.

**Microcontroller**

Voor de selectie van geschikte microcontrollers heb ik eerst onderzocht welke merken en typen veel worden gebruikt in de automotive industrie en waarom juist deze microcontrollers worden gekozen.

In de automotive sector worden microcontrollers van Infineon, NXP, Texas Instruments en Renesas veel toegepast. Deze microcontrollers worden geselecteerd op basis van de volgende criteria: ondersteuning voor RTOS, deterministische reactietijd, ISO 26262 ASIL-B certificering, ondersteuning voor automotive communicatieprotocollen zoals CAN communicatie. Die toepassingen zijn voor mij niet allemaal noodzakelijk, op de ondersteuning voor RTOS na. Hierdoor wijk ik af van de traditionele en verplichte eisen die gesteld worden aan de microcontrollers.

Ik heb aan de microcontroller de volgende voor het project belangrijke eisen gesteld: CPU snelheid en architectuur, geheugen (RAM & FLASH), RTOS, I/O, SPI/I2C/UART, DMA en voldoende ondersteuning van commodity’s en standaard bibliotheken. Om het enigszins een beetje overzichtelijk te houden heb ik de volgende microcontrollers in een tabel gezet en beoordeeld op de eisen:

* Arduino Nano ESP32-S3;
* Teensy 4.1;
* STM32F4;
* NXP KL25Z;
* Rasp Berry PI.

Zie hiervoor bijlage C dat achterin dit document is opgenomen.

Ik heb uiteindelijk gekozen tussen de twee beste de allrounder de Arduino Nano ESP32-S3 en de Teensy 4.1. De microcontrollers ontlopen elkaar niet heel erg. Wat de doorslag heeft gegeven is dat ik bekend ben met het concept van de Arduino, dus is voor mij de keuze gevallen op de Arduino Nano ESP32-S3. Ondanks dat ik Teensy 4.1 een goede microcontroller vindt, met veel mogelijkheden vindt ik de beschikbare tijd voor het afstuderen te weinig.

**Afstandssensor**

Voor de afstandssensor heb ik gezocht naar een vergelijkbaar model als het exemplaar dat momenteel wordt gebruikt in de DAQ-module. Op basis van die functionele eisen heb ik de volgende minimale technische specificaties opgesteld:

* Aansluiting: M12, 4-polig.
* Beschermingsklasse: Minimaal IP67.
* Meetbereik: Minimaal 1 meter.
* Uitgangssignaal: digitaal (PNP/NPN).
* Montage: Moet in een bepaalde hoek geplaatst kunnen worden.
* Kosten: Maximaal 300 euro (om het enigszins betaalbaar te houden)

De belangrijkste vereiste is dat de sensor betrouwbaar moet reageren op een reflectiepaaltje, ongeacht de lichtomstandigheden – zowel bij fel zonlicht als in volledige duisternis.

Om de juiste technologie te selecteren, heb ik radar, rood licht, infrarood en ultrasoon met elkaar vergeleken. Hieruit blijkt dat radar en infrarood de meest geschikte opties zijn. Radar is echter te complex en te duur, waardoor de keuze valt op infraroodtechnologie als de meest efficiënte en haalbare oplossing.

Een afstandssensor is gekozen om valse detecties te minimaliseren, een belangrijke eis van de stakeholders. Deze sensoren kunnen specifiek worden ingesteld om alleen objecten binnen een bepaalde afstand te detecteren, wat zorgt voor betrouwbare en nauwkeurige metingen. Belangrijk hierbij is dat het reflectieobject zich zo goed mogelijk haaks op de sensor bevindt. Om de meetnauwkeurigheid te optimaliseren, moeten zowel de afstandssensor als het reflectieobject waterpas worden gemonteerd. Dit zorgt ervoor dat de laserstraal optimaal wordt teruggekaatst en voorkomt meetfouten door een verkeerde hoek.

Het aanbod aan afstandssensoren is zeer uitgebreid, waardoor een gerichte selectie noodzakelijk was om de keuze eenvoudiger en overzichtelijker te maken. Daarom heb ik een selectie van vier sensoren gemaakt en deze beoordeeld aan de hand van de gestelde eisen:

* ifm O5D150
* Leuze DMU412B-500/LCT-M12
* Contrinex DTL-C55PA-TMS-619-506
* Sharp GP2Y0A41SK0F

Zie hiervoor bijlage D tabel 1 dat achterin dit document is opgenomen.

Uit deze vergelijking komt de ifm O5D150 als beste keuze naar voren. Deze sensor voldoet aan alle gestelde eisen en heeft bovendien een stevige, robuuste behuizing, wat bijdraagt aan de betrouwbaarheid en duurzaamheid in industriële toepassingen.

De ifm O5D150 vereist een externe voedingsspanning van 24V DC, wat hoger is dan de 3.3V die de Arduino Nano ESP32 kan leveren. Daarnaast moet het digitale uitgangssignaal worden aangepast om compatibel te zijn met de 3.3V logica van de Arduino.

Afhankelijk van de gekozen uitgangsmodus van de sensor PNP of NPN.

**RTC**

**Voor een probleemloze integratie en betrouwbare prestaties op de Arduino Nano ESP32-S3 is de DS3231 een uitstekende keuze. Deze RTC is niet alleen zeer nauwkeurig, maar wordt ook volledig ondersteund door de** [**RTClib**](https://github.com/adafruit/RTClib)**, die compatibel is met de ESP32-S3. Dit maakt de implementatie eenvoudiger en zorgt voor betrouwbare tijdregistratie in je projecten.**

**Open Smart DS3231 RTC Module met EEPROM**

**Display**

**geheugenmodule**

**GPS-module.**

Aanpak

*Bepaal de benodigde sensoren en vergelijk de bestaande hardwareopties.*

De sensoren die noodzakelijk zijn: Afstandssensor, RTC, Display, geheugenmodule en GPS-module.

Bronnen & Methode

*Technische documentatie, datasheets, benchmarks van bestaande modules.*

**7.2 Softwarearchitectuur**

**Tweede subonderzoeksvraag:**

*Welke software-architectuur zorgt voor een efficiënte verwerking en opslag van tijdstempels?*

Aanpak

*Onderzoek softwarestructuren voor real-time verwerking. Kijk naar embedded programmeertalen en opslagformaten.*

Bronnen & Methode

*Vergelijkbare open-source software, embedded ontwikkelplatforms (Arduino, STM32, ESP32, KL25Z).*

**7.3 Prestatie-eisen**

**Derde subonderzoeksvraag:**

*Wat zijn de prestatie-eisen voor een nauwkeurige tijdsvergelijking?*

Aanpak

*Definieer minimale precisie en resolutie die nodig is.*

Bronnen & Methode

*Wiskundige modellen, specificaties van verkeersregelinstallaties.*

**7.4 Opslag en verwerking**

**Vierde subonderzoeksvraag:**

*Welke opslag- en verwerkingsmogelijkheden zijn nodig voor logging?*

Aanpak

*Bepaal of SD-kaart, interne opslag of externe interface nodig is.*

Bronnen & Methode

*Specificaties van logging-methodes, benchmark van opslagopties.*

**7.5 Validatie en testplan**

**Vijfde subonderzoeksvraag:**

*Hoe kan de datalogger worden getest en gevalideerd op nauwkeurigheid?*

Aanpak

*Ontwikkel een testplan: welke experimenten moeten worden uitgevoerd?*

Bronnen & Methode

*Literatuur over validatiemethodes, bestaande testopstellingen.*

**8. Conclusie en Synthese**

Zodra alle subvragen beantwoord zijn, kun je ze samenvoegen om **de hoofdonderzoeksvraag te beantwoorden**. Dit leidt direct tot je **ontwerpvoorstel**.

💡 **Denk hierbij aan**:

* Welke **hardware/software combinatie** is het meest geschikt?
* Hoe kan de **foutgevoeligheid** van de BM22 worden verminderd?
* Hoe zorg je ervoor dat de **logging betrouwbaar en reproduceerbaar** is?

***Opzet***

Tijdens het onderzoek heb ik naar een aantal factoren gekeken hoe mijn ideale prototype eruit komt te zien.

Wat voor mij de belangrijkste punten zijn is:

* Soort microcontroller, besturingssysteem, programmeertaal, snelheid en geheugen;
* Meetnauwkeurigheid apparaat;
* Sensor (module of component);
* Type en wijze van opslag;
* Levensduur, onderhoud en kosten.

**Soort microcontroller, besturingssysteem, programmeertaal, snelheid en geheugen**

# Verwijzingen

*https://www.logmore.com/post/what-is-data-logging*. (sd). Opgehaald van https://www.logmore.com/: https://www.logmore.com/post/what-is-data-logging#:~:text=Data%20logging%20is%20the%20process,regardless%20of%20the%20method%20used.