Usability and Maintainabilty of a hardware project

A driving platform for all students

Date

16-11-2018

Author

Jip Galema (1679915)

Teachers

Huib Aldewereld

Arno Kamphuis

Wouter van Ooijen

Organisation

University of Applied Sciences Utrecht

Version 0.2

MANAGEMENT SUMMARY

INDEX

1. THEORETICAL FRAMEWORK

When looking at the main and sub-questions there are three concepts that will be central to this research: Usability, maintainability and the driving platform.

1.1 Usability

The meaning of the term usability within the software development standard ISO25010 consist of the following definition: “Degree to which a product or system can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use.” (ISO25000, n.d.)

As part of this definition the following sub-characteristics are defined:

- **Appropriateness recognizability:** Degree to which users can recognize whether a product or system is appropriate for their needs.
- **Learnability:** Degree to which a product or system can be used by specified users to achieve specified goals of learning to use the product or system with effectiveness, efficiency, freedom from risk and satisfaction in a specified context of use.
- **Operability;** Degree to which a product or system has attributes that make it easy to operate and control.
- **User error protection:** Degree to which a system protects users against making errors.
- **User interface aesthetics:** Degree to which a user interface enables pleasing and satisfying interaction for the user.
- **Accessibility:** Degree to which a product or system can be used by people with the widest range of characteristics and capabilities to achieve a specified goal in a specified context of use

When adjusting the usability definition to the specific context it gives a better picture of what the concept should accomplish, the specific definition for usability within the project would be:

The degree to which the platform (and its software) can be used by students to make hardware/software modules with effectiveness, efficiency and satisfaction within the R2D2 and BB8 projects.

The main focus of this research will be on enhancing the learnability and operability, these two characteristics are within the scope of the project.

1.2 Maintainability

The definition of the term maintainability within the software development standard ISO25010 is: “The degree of effectiveness and efficiency with which a product or system can be modified to improve it, correct it or adapt it to changes in environment, and in requirements.” (ISO25000, n.d.)

This definition can be divided into 5 sub-characteristics:

- **Modularity.** Degree to which a system or computer program is composed of discrete components such that a change to one component has minimal impact on other components.
- **Reusability.** Degree to which an asset can be used in more than one system, or in building other assets.

- **Analysability.** Degree of effectiveness and efficiency with which it is possible to assess the impact on a product or system of an intended change to one or more of its parts, or to diagnose a product for deficiencies or causes of failures, or to identify parts to be modified.
- **Modifiability.** Degree to which a product or system can be effectively and efficiently modified without introducing defects or degrading existing product quality.
- **Testability.** Degree of effectiveness and efficiency with which test criteria can be established for a system, product or component and tests can be performed to determine whether those criteria have been met.

When adjusting the maintainability definition to the specific context it gives a better picture of what the concept should accomplish, the specific definition for usability within the project would be:

The degree of effectiveness and efficiency with which the platform and software can be modified to improve it, correct it or adapt it to changes in environment, and in requirements.

The main focus to enhance the maintainability will be on modularity and reusability, these two characteristics are within the scope of the project.

As part of this research the different decisions that come with increasing the degree of the sub-characteristics are reasoned and documented, this will give a clear view of the type of platform that needs to be build and how the software needs to be structured. Usability will always come first since after all this is still a project for the near future.

1.3 Driving platform

The driving platform is the concept that brings both concepts mentioned above together. The driving platform will have motors and wheels and a platform to connect and place electronics on, and while it might not be as important as the other two concepts it is still where most of the usability and maintainability is based on.

The goal of the platform is to meet the requirements set in part of the MoSCow analysis on page 10.

Important literature

A big part of this research will be looking through papers that have tackled similar problems. Papers can be found on different databanks like:
- Google Scholar
- Web of Knowledge

Some of the relevant search terms that can be used in these databanks are: driving platform, reusable hardware library, usability in software. Other terms that are relevant to the research and can be used are: platform, hardware, reusability, maintainability, robot, ISO25010

The other part of the research is working through the code of other libraries that are being maintained throughout the years to find different characteristics that make up such a library.

2. The Assignment

The goal of the assignment, as explained in the introduction, is to design and build a driving platform that can be used for educational purposes within the bachelors Applied Artificial Intelligence and Computer Engineering. The most important thing is that the platform is usable first, and maintainable second.

2.1 Phases of the project

As part of the assignment there will be three phases: Analysis, design and realisation.
The analysis phase will be focused on the hardware/software that increase the usability and maintainability of the driving platform and the decision that must be made to get to an architecture. This architecture will consist of the different hardware parts that go into the platform and their arguments.

The second phase is the design phase, this phase will focus on researching the software part of the platform. After the architecture is set there will be a lot of software that has to be made. The requirements of the software will be somewhat vague, for example: make sure the microcontroller can communicate with the motors. These requirements will be finetuned in this phase. As part of this phase there will be research into making these hardware/software libraries usable and maintainable. This phase will also be used to test this code on the hardware. At the end of the phase there will be a blueprint. This features the architecture as well as the structure of the software and the working of it. The blueprint will feature class diagrams and STD's to give a clear view of the project architecture
The last phase is the realisation phase. After the first two phases there is an architecture and a blueprint. From this architecture a big part of the code has to be made and tested. The only thing left to do is to discuss with the clients if the requirements are met and build the platform. This will be mainly a practical phase and there is very little opportunity or need for more in-depth research. At the end of the phase there will be a platform and a manual for constructing the platform and a documented hardware/software library.

The following items must be delivered:

- Platform
- Documented code library
- Research plan
- Research
- Architecture document (list of hardware and software components)
- Assembly manual

2.2 Project method

This project will use the agile method. This is ideal since there needs to be a lot of feedback from the clients in how the platform is going to look hardware and software wise.

2.3 Main question and sub question

To make this driving platform there needs to be some research on the hardware/software components. To understand the terms used in the sub-question there will be a glossary below:

Architecture: A document of components that are used in the driving platform with the arguments to use these components

Blueprint: A document with tested components and a thought-out structure for the code, as well as an system architecture. The only thing that has to happen after this document is connect the different code modules and assembling the platform.

The main research question will be:

> What is the driving platform blueprint that featured usability in current second-year project, for the bachelors Computer Engineering and Applied Artificial Intelligence, and maintainable for the next years and other projects?

To answer the main research question there must be sub-questions, these are below:

- What is an architecture for a driving platform that is both usable and maintainable?
- What is a blueprint for a hardware and software library that is both usable and maintainable within the context of the driving platform?
- What aspects of the blueprint live up to the standards of the clients, and what aspects don’t?

2.4 Changes

During this research there was a shortage of both time and specific resources, which means part of the research could not be completed. As explained in the assignment plan[1] this was one of the risks. Because of horizontal slicing the research will focus around the hardware architecture and software architecture. While the product review will not be included, the problems that appeared due to the hardware will be explained in the hardware architecture.

[1] See appendix 1

3. Hardware architecture

Before designing the platform and its software the different hardware components have to be chosen. The different choices of components are explained below.

3.1 Microcontrollers

The following microcontrollers are up for consideration for this platform:
- Arduino Due
- Raspberry Pi 3B+
- Pine64 board
- Esp32

All these boards are available at the Turing lab which makes it easy to test. The list is quite specific mainly to control the scope of the project, and available to use immediately, there are microcontrollers that maybe better suited for this platform, but also require more research that would significantly increase the scope of the project.

Arduino Due

Arduino is an opensource-computer platform that can be used to make microcontrollers, they come in all shapes and sizes. While there are other languages that can be used to program Arduino's the preferred language in either C or C++. The Due specifically has an ARM-chip at 84 MHz and 512 kB of flash memory. In the official Arduino store the due costs 35 euros, by buying a 'remake' the price can get down to 14 euros.

Raspberry Pi 3B+

When looking for a microcontroller that has more power than an Arduino a Raspberry Pi comes to mind. The Raspberry Pi Foundation provides low-cost, high-performance computers. The Raspberry Pi's are powerful enough to have an operating system and is able to run a full python version. The Raspberry Pi 3 Model B+ has a 64-bit Quad-core processor at 1.4 GHz and 1 GB of RAM.
The average price for a Raspberry Pi 3 Model B+ is around 35 euros.

Pine64

Pine64 is a company that makes computer boards that feature a 64-bit chip, these chips are way more powerful and can run on Linux. There is a Quad-Core ARM Cortex A53 64-Bit Processor and 2 GB of RAM. While the company makes individual board for hobbyist, they also sell laptops with the same chips. The prices are comparable to those of the Raspberry Pi's at about 32 euros.

Esp32

The esp32 is a low-cost Wi-Fi chip that can be used as a microcontroller. The esp32 works with the Arduino firmware and with MicroPython (an implementation of Python for embedded devices). The esp32 uses a 32-bit dual-core processor operating at 160 or 240 MHz and has 520 kb SRAM (RAM that needs to be periodically refreshed)

3.1.1 The decision

The Microcontroller requirements for the different bachelors are mainly focused on the difference in programming skill that exists within the different bachelors. For example, the bachelor computer engineering (CE) has a focus on embedded programming, this is mainly done with low-level programming languages like C++ and C. The bachelor Applied Artificial Intelligence (AAI) is focused on algorithms.
The focus is more on the mathematical side of the algorithms, when programming does happen this is done in more high-level languages like python. While CE is about interfacing (talking) with the hardware on low-power and low-memory microcontrollers, AAI is focused on setting up algorithms that work on PC's. In both cases, the main objective to write efficient code.
As defined in the theoretical framework, the two characteristics, the focus of this research will be usability and maintainability. As part of enhancing both characteristic, the two sub-characteristics are evaluated for every need within the subject of microcontrollers.

3.1.2 Usability

The two sub-characteristics that this research focusses on are learnability and operability.

Learnability

As can be found earlier in the theoretical framework the definition of learnability within the ISO25010 standard is: Degree to which a product or system can be used by specified users to achieve specified goals of learning to use the product or system with effectiveness, efficiency, freedom from risk and satisfaction in a specified context of use. Within the first two years of CE there are a few libraries, tools and languages that students have been using:
- hwlib: An Object-Oriented library for Arduino Due.
- bmptk: Builtenvironment for C/C++/Assembler and microcontrollers.
- Cortex M0 assember: Machine language for Arduino.
- Rtos: real time operating system for Arduino due.
- C++.
- Python.
- Arduino Due.

When looking through the list there is one common microcontroller that has been used by every second-year student. This is the Arduino Due. While there are students out there that have worked with other microcontrollers, this is not required and usually means that it is done as a hobby. Concerning the learnability, it is important for students to use a software library they are familiar with. The CE-students rarely use the Arduino IDE, the hwlib-library (mentioned above) is used in the semester before the actual R2D2 project. The hwlib-library only supports the Arduino Due and Arduino Uno (a less powerful version).

For AAI student the curriculum states that the following libraries, tools and languages are used:
- Raspberry Pi(optional, the first year project requires everyone to work on this part)
- Python with numpy and pandas
- Object-Oriented programming in either Java, C++ or Python

While the whole curriculum is not yet set (see the second point for example) there is a clear difference between the two bachelors. The AAI-student program on higher-level languages then CE-students. The curriculum states that there is no course that specifically uses a microcontroller. To increase the learnability, it is important to use a well-documented microcontroller that can easily be used, since some of the students have already used a raspberry pi this is the preferred choice.

Another option to accommodate the need for a Raspberry Pi but still hold the same amount of learnability of the CE-students, is by making a hardware library that holds similar syntax to that of the hwlib library. This way the students can upload the same type of code just on another platform.

Accessibility

The definition of accessibility within the ISO25010 standard is: Degree to which a product or system can be used by people with the widest range of characteristics and capabilities to achieve a specified goal in a specified context of use. So, while learnability is specific for the second-year CE and AAI-students. Accessibility is about other students and teachers in general. For that to be doable it is important to use microcontrollers that are easy to use for people that have never worked with one before. A good example of a microcontroller that can be used by a lot of different people is the Raspberry Pi. The first year of all bachelors is based around a common set of courses. This means that all students (even those that are studying Business IT and Management) have to work with a Raspberry Pi for 3 weeks. This shows that it is quite easy to learn how to work with a Raspberry Pi without having any experience.

3.1.3 Maintainability

The two sub-characteristics that this research focusses on concerning the maintainability within the ISO25010 standard are modularity and reusability.

Modularity

As told earlier the definition of modularity is: The degree to which a system or computer program is composed of discrete components such that a change to one component has minimal impact on other components. Are the microcontrollers usable in other instances, is it economically viable for the university, is the best way to interpreted this. Almost all microcontrollers can be used in some other way shape or form, therefore the characteristic has no real influence in the decision of which microcontroller to use.

Reusability

The definition of reusability is: Degree to which an asset can be used in more than one system, or in building other assets. The definition is like modularity. Just like with modularity the microcontrollers could be separated from the platform, there are a bunch of uses for microcontrollers that can be used when not on the platform.

3.1.4 Conclusion

The curriculum for the different bachelors presents some challenges since both benefit from different microcontroller, CE benefits from the Arduino Due and AAI form the Raspberry Pi. As told in the learnability there is however a way to satisfy both bachelors with one microcontroller. This however would require a new implementation of the hwlib library currently used by second year CE-students. Concerning the accessibility, the Raspberry Pi is often described as a microcontroller that can be used for beginners. The accessibility of the Raspberry Pi is on full display when looking at the alarm systems even Business IT and Management students can make with the microcontroller without having any past experiences with the Raspberry Pi or programming at all. While maintainability is important the microcontrollers described above are in no way, shape or form something that can only be used within a driving platform.

As described above the decision comes down to either the raspberry pi or the Arduino due. For AAI students it is important that the microcontroller can be used to execute algorithms and has some computing power. Since it is possible to program with C++ on the Raspberry Pi that is also easiest to learn for all students. Since the price of the platform would be significantly higher there doesn't seem the need to add an Arduino due board or use one in general. To accommodate both bachelors it is important to make an hwlib-like library that works with C++ on the Raspberry Pi, ultimately the syntax should be the same, so the learnability is not affected for CE-students.

3.2 Motors

For the platform to be a driving platform there need to be wheels and motors. There two requirements for the movement of the platform. First being able to turn a certain amount of degrees (with a maximum offset of 5 degrees) and secondly drive a certain distance. Before getting into the choices that are the best for the usability and maintainability of the platform, it is important to know which motors are available and how they work.

For this research the following three different types of motors will be evaluated: DC motors, servo motors and stepper motors.

DC-motor

A DC-motor (DC stands for direct current) has 2 wires (power and ground). When power is applied to the motor it will start turning. When the power is cut off the motor will stop. The amount of revolution per minute (RPM) gives an idea of how fast the motor is. DC-motors are easy to use with things that need to turn fast. There is no direct way to measure the amount of rotations within the motors, but it is possible to connect an add-on that gives the feedback.

When researching different driving platforms, the common motor used was a DC-motor. These are often powerful enough to transfer larger weight. With this power often comes a motor that heavier than the other mentioned below. The price-tag for DC-motor is on average higher than the stepper motor and servo[2].

H-bridges

While DC-motors are relatively easy to control there is still a controller needed between the motor and the microcontroller. Generally, this controller is called a H-bridge a few, in the CE-lab available, will be structure to show the different

Name	Voltage	Current limit	Price
TB6612	4.5 – 13.5V	1.2 A	€ 1, -
L298N	3.0 – 14V	5A	€ 1.70, -
IBT_2	6 – 27V	43A	€ 13, -

Servo motor

A servo motor has three wires to connect with a ground, power and control. The control receives the position of the servo. Usually servos have a rotation of 90 or 180 degrees. There occasionally a servo that can rotate 360 degrees, but a servo does not rotate constantly. Therefore, in this context of the servo will be of no use.

[2] https://www.pololu.com/category/22/motors-and-gearboxes

Stepper motor
A stepper motor is fundamentally like a servomotor. The only difference is that there is no continuous rotation, but every rotation is a small step. The main advantage of a stepper motor is that it is easy to setup and is precise (every rotation is generally 0.6 degrees). To resist movement into the wrong direction the stepper motor constantly draws its full current. To increase efficiency the stepper motor can be connected to a stepper motor driver board. Driver boards also limit the amount of current to increase power efficiency.
An important part of choosing a specific stepper motor is determining the holding torque, which is the force the motor can hold when in a certain position. The higher the kg-cm the more power a stepper motor has.

While it is an advantage that a stepper motor doesn't need an encoder to know the amount of rotations it is also one of the problems. In a situation where the motor is blocked while trying the rotate (for example someone holding the wheel) the motor cannot register that the rotation has not taken place.

The following weights in the platform need to be considered:
- Raspberry Pi 3 Model B+, 50 grams
- Some kind of platform, max. 200 grams
- Initially two Breadboards (gives students the freedom to make their own circuits, 150-200 grams
- 3D Printed wheels and motor holders, 50 grams
In general, the platform is relatively light at around 600 grams in basic weights. The motor doesn't need to be as powerful, since there are two in the platform. About 1kg/cm of holding torque would be enough to move the platform forward at a certain speed. The calculations that decided the torque are not as simple as the calculation above, but due to the scope of the project we will not get into this calculation

Stepper motors are generally not found in driving platforms but can be a viable option since the platform in this context will be quite small. The important difference with the DC-motor mentioned above is the price. A stepper motor with 1kg/cm holding torque is around 6-8 euros. While a matching driver platform are about 6-7 euros. While on most sites DC motors with encoders are slightly more expensive than stepper motors with the same torque. When doing some more research there was however a great deal for DC motors with encoders and wheels. This set was combined 24 euros which is significantly cheaper than buying stepper motors (12 euros a motor) with drivers (6 euros a driver)[3]

3.2.1 Usability
The definition of the two sub-characteristic is still the same as with the microcontrollers.

Learnability
Learnability, as told earlier, is about how easy it is for student to learn how to use the technology. If the code designed in the next chapter is correctly implemented the difference between the learnability shouldn't be influenced, since the use of the motor and its functions should be the same.

[3] https://www.aliexpress.com/item/1-set-2-set-Mayitr-DC-6V-210RPM-Encoder-Motor-Gear-4mm-Shaft-With-Screws-Mounting/32823003082.html?spm=a2g0s.9042311.0.0.27424c4drpFsLZ

Accessibility
The accessibility of motors is mostly about the software. Considering that most of these users are not familiar with programming they will probably not get into the difficulty of making a hardware library. For the software it is important that it can both be specific and general at the same time. People that have not programmed before should be able the send easy commands like:" drive forward 10 meters" while second CE and AAI students might want some more control for example:" Turn wheel-A 30 degrees".
While most of this will be addressed in the system architecture one hardware components that needs to be available in the motor is; a feedback mechanism. Because the surface of the platform is often different and gives different type and amount of resistance it is very unpredictable to take a certain amount of time that the motor is on that can relate this to a distance. There are two types of motor that have both continuous rotation and a feedback mechanism, these are a DC motor and stepper motor.

3.2.2 Maintainability
Just like with the usability of the motors, the same definition is used for the two sub characteristics as with the microcontroller parts.

Modularity and reusability
The motor and its library should be able to be used in different systems. Just like with the accessibility this is mostly be about software. The reusability is about the same. The difference is that not every type of motor is being used within the school. There are a few examples of servos being used by first year student to make robotic arms. Second year students have used stepper motors to make a robotic arm. So, there is enough use for either a servos or stepper motor, but DC-motors can be used in other projects as well.

3.2.3 Conclusion
As told above DC-motors are often used in driving platforms but come at a relatively high cost. While stepper motors are less common but also less expensive.

The stepper motor best suited for this task is a NEMA14 stepper motor these need 12 Volts and 0.4A/Phase per coil, there are two coils so 0.8A/phase. One of the advantages of these motors is that school already has a pair of motor drivers that can be used for testing and managing the motors. The only motor drivers that support 12 volts power and 0.8 A of current are the DRV8825 stepper motor drivers.
The DC-motor best suited for this platform would be the JGA25-370 DC gearbox. This motor has an input voltage of 6V and a stall current of 900 mA (the maximum current drawn) and a free current of 80 mA when the motor is rotating freely at maximum speed, with on load. The motor has 5kg-cm holding torque. On the back of the motor an encoder is attached. As can be seen in figure 2.1 the pin has 2 plus- and min-pins. And two phase pins that can be used to receive feedback. The 'driver' to control both motors will be a H-bridge which is also available in the CE-lab.

Figure 3.1 DC motor encoder pinout

Considering all things, the best motor to use is the DC motor this mostly has to do with the price and usability. The set with two DC motors, encoders and driver are cheaper than 2 stepper motor with drivers. As told before the stepper motor can be slightly off about rotations when blocked, this is not the case with DC motors. The H-bridge will be the L298N, this is mainly because of the higher current limit. Since the motor require a relatively high current (900 mA times 2) it is important for the drivers to able to handle this amount.

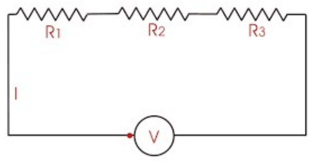
3.3 Power supply

Before getting into the different kinds of power supplies, there are a few hardware requirements concerning power that apply to the different hardware on the platform:
- Raspberry Pi 3 Model B+: 5V at 700mA-2.5A.
- 2 DC motors at 6V 900mA

As can be seen there are two types of voltages and current there are some changes that need to be made to prevent an accident. The best way to do this is by reducing the voltage and current that go to the different components by connecting components in series or parallel.

Series

When connecting things in series the current over the entire circuit stays the same while the voltage drops with every resistance the electrons encounter in the circuit.



Parallel

A parallel circuit is the opposite, the path of travel is divided, this is why the voltage is the same and the current divided over the number of paths.

While these are the two fundamental options to divide voltage and current in a circuit. There is hardware available to increase or decrease the voltage within a circuit.

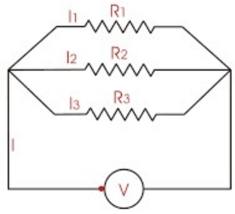


Figure 3.2

DC-DC converters

Since there is a difference in voltage between the different hardware components some sort of conversion needs to happen. One way to do this is by using a DC-DC booster. There are two type of DC-DC boosters: step up and step down. A step-up DC-DC booster board converts voltage to a higher voltage, while a step-down converts voltage to a lower voltage.

Most converters have about a 94-96 % efficiency rating this means that the voltage and current will never be precisely what is expected. For this reason, it is important to calculate some extra current and voltage when calculating with these boosters.

While in theory these boosters work well with low voltages this should be taken with a grain of salt. When testing two different boosters (see test reports in appendix 2) both the input- and output-current/voltage where put thought the test and findings where not as predictable as was expected. The first booster converted 3V to 5V but only gave a 700 mA current when using an input of 2A. While boosters that can boost up to 30 V with output currents as high as 6 A won’t work below 4 volts.

Because of the scope of the project, and the findings in the test report, this research will focus on both LiPo batteries and rechargeable AA batteries.

LiPo Batteries
LiPo batteries (Lithium Polymer batteries) are often much lighter and have higher capacities (allowing them to hold much more power) and higher discharge rates (they pack more punch, displayed with the C rating). While these batteries often have more power some of the downsides are a shorter lifespan and difficulty with charging, when using a multi-cell battery. A single cell can supply an average voltage of 3.6 Volts (Usually the voltage fluctuates between 4.2 and 3.2. The current is often depended on the capacity of the battery, this capacity is often given in mAh. This means a battery with 2000mAh can supply 2 Amps of current for one hour. Whether a battery can reach this amount of current is depended on the C-rating. A battery that has a rating of 1C discharges the whole battery current in 1 hour. The higher the current the shorter the time it takes for the battery to discharge.

Rechargeable AA batteries
AA batteries come in different capacities, voltages and chemical elements. There are nickel-cadmium (NiCd), zinc-carbon, Zinc-chloride, Alkaline (Zinc and manganese dioxide), nickel-metal hydride (NiMH) and Li-Ion.
A number of these batteries are rechargeable, these are: NiCd, NiMH and the Li-Ion batteries, some of the Alkaline batteries are also rechargeable. All of them have different capacities and voltages as can be seen in figure 1.2.

Type	**NiCd**	**NiMH**	**Li-Ion**
Capacity	600-1000 mAh	600-2850 mAh	9900 mAh
Voltage	1.2V	1.2 V	3.7 V
Price (per battery)	€ 1.30	€ 1.40	€ 1.20

Looking at the table the Li-Ion batteries have the best capacity and voltage and are slightly cheaper than the other option. This would be a good option to use in the platform because of the flexabilty the cells have and how they can be connected in series as a multicell battery but individually charged.

The following requirements need to be met for the batteries to be both usable and maintainability:

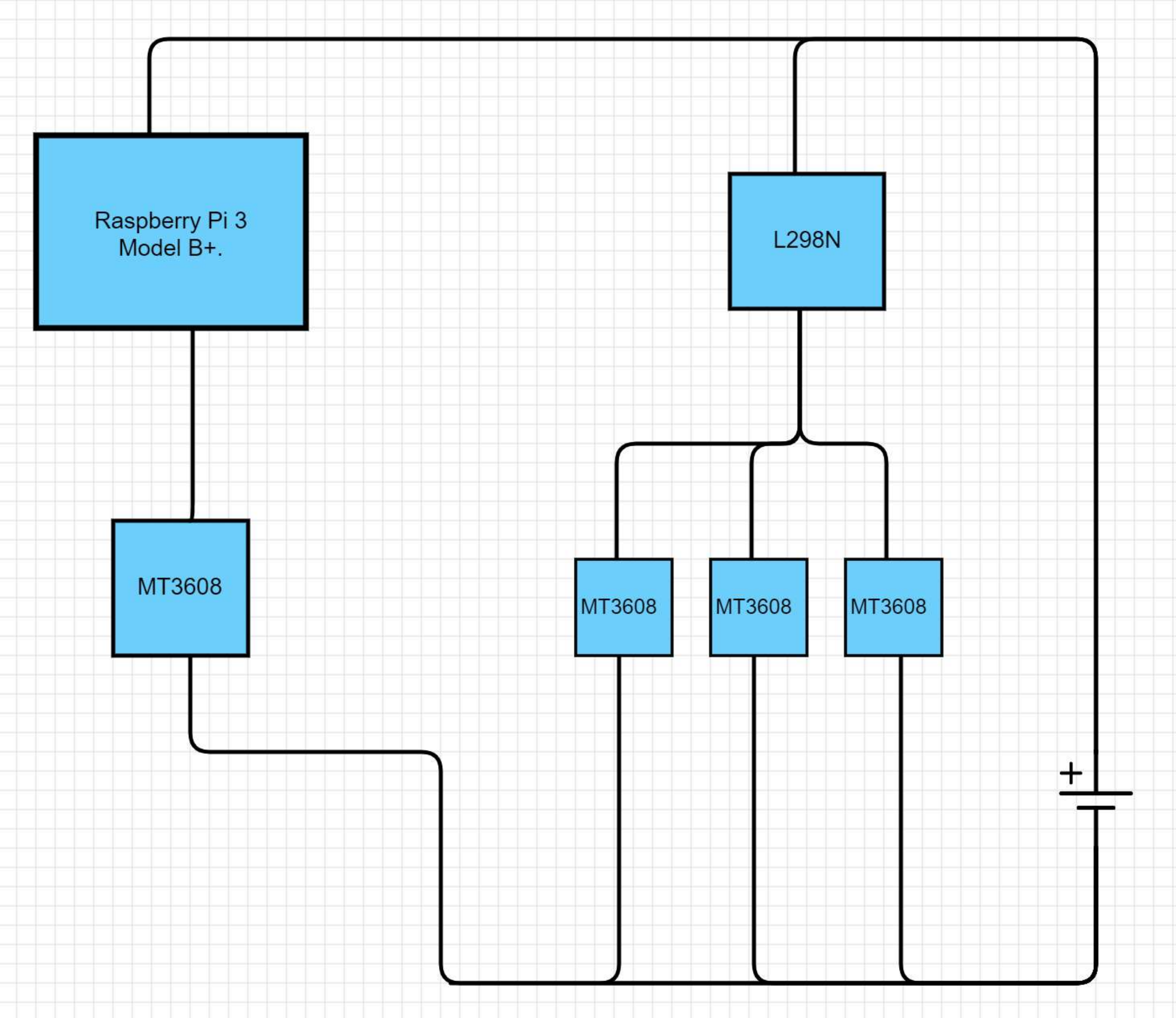
3.3.1 Usability
The batteries must be easy to use. This means that it should be easy to connect the batteries or charge the platform. For testing to be easy the platform needs to be able to drive continuously for 30 minutes, as that is a requirement set by the CE-department. Since the minimal current needed is about 4-4.5Amps. A battery of 4000mAh is plenty enough to last an hour, the perfect scenario would be a 2000 mAh with a 2C rating. But when looking into the availability of such batteries this was either out of stock or more expensive than the 4000 mAh 1C batteries available. The usability requirement means that a battery should have a capacity of 4000 mAh and a 1C rating.

3.3.2 Maintainability
The batteries need to be easy to charge either while still attached to the raspberry or connected to a power station. The use of rechargeable batteries is key. While a single cell (one battery) is easiest it means being stuck at a 3.7 voltage, with individual batteries cells it is easy to customize the current and voltage, but one cell is simple to charge. Since there are voltage boosters that can convert a low current to a high current, it should be possible to convert 3.7 to 5V and 6V. This means that the single cell battery appears to be enough for our purposes.

3.3.3 Conclusion

As explained earlier the components have different voltages. To convert the current voltage (3.7V) to 5 and 6V there will be one type of booster. A MT3608 converts 2-24V to 5-28V with a maximum output current of 2A. This makes it possible to use a single cell LiPo-battery. Because the motor requires 900 mA and the booster gives 600 mA when working with 6V output there will be three boosters connected in parallel. This combines all currents to 1800 mA, which should be enough to fully power the motors.

The cell has 3.7V of power and 8000mAH capacity with a 1C rating. This is to gives all boosters enough current to make the conversion. The following schematic will be used to connect all the components



Due to the lack of time the parallel boosters were not tested in this context. This testing will be part of the recommendation at the end of the research.

3.4 Other components

There are a few small components that do not require a lot of analysis like the other components do.

3.4.1 Platform

Both the wheels and holder of the motor will be 3D printed. This is mainly because it saves money and makes sure the components can be made within the premises of the University. An alternative to 3D printing would be buying premade stepper holders and wheels made by the vendor of the stepper motors. But since these are relatively expensive working out a way to print the part, is more cost-effective. The actual platform will be either 3D-printed or cut out of wood from the machines currently in use on the second floor of the Heidelberglaan 15.

The two wheels, at the back of the platform, will be connected through a metal bar and placed so they can rotate but don't add any motor power. To still live up to the rotation precision requirement both stepper motors (turning in opposite direction) can be used to increase the accuracy of the turn. As told in the beginning of this research due to a lack of time this part of the hardware cannot be tested yet, so no examination can be made on if this is the best way to construct the platform.

While four stepper motor would help with the precision this is also twice as expensive.

3.4.2 Other sensors

There will be some other sensors connected to make it easier for AAI students to work with the platform. This will either be a camera, four ultra-sonic sensors or touch sensors that can be used for position-calculations and autonomous driving projects.

3.5 Final component list

Rounding everything up the following components will be used in the platform:

- Raspberry Pi 3 Model B+
- JGA25-370 DC gearbox with encoder, 6V – 900 mA
- L298N H-brigde
- 2 MT3608 DC-DC Booster step up 2V-24V to 5V-28V, 2A
- Lithium battery 3.7V 8000 mAh.
- 4 3D printed wheels
- Metal axle bar
- Wooden platform
- 2 3D printed motor holders
- Screws and bolts

4. Software Architecture

As part of the learnability of the driving platform there will be implementations in both python and C++ to control the different types of sensors on the platform and the motors of the platform. The different libraries and structures of the two implementations are discussed below.

4.1 C++

To accommodate the CE-students there will be a C++ implementation that has the same syntax as the current hwlib-library. As mentioned before the learnability and maintainability is greatly increased by using the hwlib library already known to all CE-students, this will be the plan for the C++ implantation.
Before looking at the different options that can be used to achieve this fact there will be a short analysis of the library, structure and tools needed to use this library.

4.1.1 Hwlib

The current library has support for Arduino Due, Db103, Arduino Uno, stm32f103c8, Kl27z, windows (only dummy pins) and pyd. The different devices use a general part of the library, while implementing their own hardware specific things. In the case of the stm32 and Arduino Due there is a difference in the calling of pins, time and using UART (**short voetnoot explain)**. Both microcontrollers have a file that implements these differences. The hwlib.hpp file then decides what 'extension file' it needs to use for this specific target.

```
#ifdef BMPTK_TARGET_arduino_due
   #include HWLIB_INCLUDE( hwlib-arduino-due.hpp )
#endif

#ifdef BMPTK_TARGET_db103
   #include HWLIB_INCLUDE( hwlib-db103.hpp )
#endif

#ifdef BMPTK_TARGET_arduino_uno
   #include HWLIB_INCLUDE( hwlib-arduino-uno.hpp )
#endif

#ifdef BMPTK_TARGET_stm32f103c8
   #include HWLIB_INCLUDE( hwlib-stm32f103c8.hpp )
#endif

#ifdef BMPTK_TARGET_frdm_kl27z
   #include HWLIB_INCLUDE( hwlib-frdm-kl27z.hpp )
#endif
```

Figure 2.1: Hwlib.hpp

The hwlib library consist of the following functionalities: pins implementation, i2c, spi, oled, matrix keypad, io expanders and graphics. Because the library is extremely modular all these functionalities are useable with every implementation.

As can be seen in figure 2.1 the library calls on BMPTK_TARGET to decide which microcontroller is used to run the library. What bmptk is will be discussed below.

4.1.2 Bmptk

Bmptk stands for Bare Metal Programming Tool Kit. This kut can be used for make-based development for microcontrollers. The folloriwng languages and compilers are supported: GCC C, C++ or assembler on Windows and Linux. Bmptk can be seen as the link between Hwlib and the native compiler used on either the computer (windows or linux) or microcontroller. In the case of the computer the program uses the GCC compiler to generate an executable, while with the microcontroller generates assembler.
At first glance it seems like raspberry pi supports needs to be added to be able to run on bmptk. While this could be an option there is an easier solution to the use of bmptk. Since the Raspberry Pi can run linux it can be seen as a native target. This means the GCC compiler will be used to build and compile the library, as faith would have it wiringPi also supports GCC. Because this is the superior solution it is important to implement this one first. Which would require the following steps:

Design

There are two possibilities to implement the hwlib library on the Raspberry Pi: creating a new library with the same syntax or implementing Raspberry Pi support for the current hwlib library. The first option (creating a new library with the same syntax) is a relatively easy option, it doesn't require as much research into the workings of hwlib and bmptk but creates a lot of overhead **(voetnoot),** while also decreasing the maintainability, if something is changed in the current hwlib-library it needs to be manually implemented in the new library. For this reason the second possibility (implementing Raspberry Pi support for the current hwlib library) should be considered first.

4.1.3 How

There are a few libraries that can be used to implement GPIO control on the Raspberry Pi, but because of the scope of this project the focus will be on the following two libraries:
Bcm2837 by dkja [4] or WiringPi. After testing both libraries the lack of documentation for the Bcm2837 library made it really hard to implement a simple program without having to add a lot of extra compiler build commands or manual building, compile and run the program, both decrease the uability. The wiringPi library didn't have this problem.

Wiring Pi

Wiring Pi is an GPIO interface library for Rapsberry Pi. The syntax of the library is relatively simple:

1. Run wiringPiSetup
2. Set the pinmode of the pin in question (output or input)
3. Either use analog/digital- write/read to use the pin.

[4] https://github.com/dkja/bcm2837

4.1.4 Extension file

To make the extension there are three functionalities that differ from others that need to be implemented.

Pins

There are a few different types of pins that can be implemented like: input pins, output pins, analog pins and in-out pins. While these are pins different in functionality implementation is not that

```
class pin_adc : public hwlib::adc{
    private:
        uint8_t pin;

    public:
        pin_adc(uint8_t pin):
            pin{pin}
        {
            pinMode(pin, INPUT);
        }

        pin_adc(pins pin):
            pin{pin}
        {
            pinMode(pin, INPUT);
        }

        adc_value_type get() override{
            return analogRead(pin);
        }
};
```

```
class pin_out : public hwlib::pin_out{
    private:
        uint8_t & pin;

    public:
        pin_out(uint8_t pin):
            pin{pin}
        {
            pinMode(pin, OUTPUT);
        }

        pin_out(pins name):
            pin{name}
        {
            pinMode(pin, OUTPUT);
        }

        void set(
            bool v,
            hwlib::buffering buf= hwlib::buffering::unbuffered
        ) override{
            digitalWrite(pin, v);
        }
};
```

Figure 2.2: Raspberry Pi implementation of pin_adc and pin_out

different. The similarity in implementation can be seen in figure 2.2. While pin_adc and pin_out have vastly different functionality the implementation looks somewhat similar.
The implementations for the pins in combination with Wiring Pi are relatively simple to use but aren't the only thing in need of a new implementation.

Timer

To implement the different timer functionalities that hwlib uses the now_ticks() functionality needs to be rightly implemented. All other time functions use the now_ticks function to decide different times. The now_ticks function in the case of the due gives the time since the time that the function now was called upon. While the in other implementation the time is since the start of the program. When the time start is not the problem since all function concerning time only use the difference between two now_ticks function calls. The other function that needs to be defined is ticks_per_us. This function is depended on the speed of the processor.

The time library (use include time.h) provides a function clock which gives the processor time consumed by the program **(voetnoot reference)**. Using the clock function will return a clock_t object that can be converted to an interger. The variable CLOCKS_PER_SEC is platform depended and gives the amount of ticks per second.
On the Raspberry Pi this variable is one million. This means one tick every microsecond, the return value for the ticks_per_us function us 1.

UART
Hwlib is generally used with microcontrollers attached to a computer, to allow the users the opportunity to use hwlib::cout UART is implemented. UART (Universal Asynchronous Receiver/Transmitter) is used to transmit and receive serial data. While this works well for the Arduino the Raspberry Pi works quite differently. The two ways that are used to connect the user to an interface for programming use either SSH or a HDMI-cable. In essence is SSH using the desktop remotely over a network. Because both versions use the actual operating system to program, the print statements don't need to be send out over UART. To show printed statements the standard cout statement can be used to print to the command line. This makes implementation of UART quite simple.

Bmptk
As told before the native Linux target is about the same as the Raspberry Pi. In figure 2.1 it can be seen that bmptk matches different targets to an extension file. As can be seen in figure 2.3 the target for linux consists of a chip, prefix, result, run and run_pause. Because the Raspberry Pi runs on linux these settings can be copied. The only change needs are another define name (Rapi), chip (BCM2837) and prefix. The GCC compilers is already installed on the Raspbian image used in this Raspberry Pi so the prefix is the path to this GCC compiler, which is /usr/bin/gcc. There is one extra setting that needs to be added, this is TARGET_LN_FLAGS. This is a linker option that always needs to be added when running this target. When using WiringPi the option -lwiringPi needs to be added to fully compile the library.

```
# running native on Windows
define Windows
   CHIP                 := Windows_Native
   PREFIX               := $(GCC-WIN)/bin/
   TARGET_CPP_FLAGS     += -mno-ms-bitfields
   TARGET_C_FLAGS       += -mno-ms-bitfields
   TARGET_LN_FLAGS      += -lgdi32
   RESULTS              += $(EXE)
   export PATH          := $PREFIX:$(PATH)
   RUN                  ?= ./$(EXE)
   RUN_PAUSE            ?= $(PAUSE)
endef

# assume running native on some Linux
define Linux
   #CHIP                 := Linux_Native
   #PREFIX               :=
   #RESULTS              += $(EXE)
   #RUN                  ?= sudo ./$(EXE)
   #RUN_PAUSE            ?= $(PAUSE)
endef
```

Figure 2.3 Targets for Windows and Linux

DC motor control
As mentioned before the DC motors have an encoder attached to them. This encoder can be used by implementing the Encoder library.[5] As can be seen in the library documentation and examples the position can be calculated by using the two phase pins.
The only thing that is needed after this is controlling the motors through the H-bridge. This is not as complicated as the encoder and can be done with simple pins, a lot of tutorials can be found that explain this for programmers just starting out with arduino. This library will be made independently from the hwlib library at first but when optimized for all boards can be integrated into the library if this is approved by the library's owner.

[5] https://www.pjrc.com/teensy/td_libs_Encoder.html

4.2 Python

For AAI-students the main language that is supported for the platform is python. The considerations made to come to a final design for this library are discussed below.

RPi.GPIO

The RPi.GPIO library will be the library used to implement the other pieces of hardware. This is mainly because the students have used this library before in the first year Raspberry Pi project.

Pins

The library uses two ways of numbering the IO pins; BOARD or BCM. BCM uses the numbers connected to the GPIO pins, while BOARD uses the actual pin number. The numbers are vastly different GPIO pin 14 corresponds to pin 8 on the actual header. Because there are a number of different versions of the Raspberry Pi the header changes from time to time relocating different GPIO pins, this makes the BCM version more reliable to withstand the test of different versions of Raspberry Pi's.

The setup of different pins is quite like wiringPi. First the pin must be setup. This means defining whether a pin is an output or input pin. Then a function can be used to either get the output or input. After use it is good practice to call the clean-up function. This will clean up all pins that have been used when running the program.

Pwm

The library also supports using pins for PWM (Pulse-width modulation) which can come in handy when looking at implementation for the motor. Using PWM is similar to using a normal pin. First initialize the pin using a frequency, then use start and stop. The frequency and duty cycle (the proportion of intervals) can be changed using functions.

L298N

The L293N H bridge[6] has a library that can be used for basic motor functions like turning the motor clockwise and counter clockwise. This library is well documented and has specific hardware setups to help user connect this chip. This library will be used for implementation.

Encoder

As can be seen in figure 3.1 the encoder used two phases to detriment the amount of rotations made. There are encoder libraries available, for example the py-gaugette library. The library supports rotary encoder usage but also a buch of other functionailites in python, for example: Rgb led usage, switch usage and Oled usage just to name a few. Every functionality has an example of how to use the code which makes the library learnable.

Other sensors

As explained in the hardware architecture the platform can contain a number of sensors that are used by AAI students. This could be: A camera, touch sensors and ultrasonic sensors. To increase the usability of these sensors and not completely leave the AAI students hanging. The following libraries can be used to implement the different functionalities of the sensors above.

[6] https://custom-build-robots.com/electronic/instruction-l298n-h-bridge-is-controlled-by-a-pca9685-servo-controller/8809?lang=en

Camera
A library that increase usability is the piCamera library.[7] This library is well documented and is made for usage by first time programmers with tutorials on installation and demo files.

Touch and ultrasonic sensors
Both touch[8] and ultrasonic[9] sensors are similarly generic sensors that are often well documented with tutorial that can easily be followed. Both feature schematic of how to connect sensors and example code that can be used for implementation.

4.2.1 Structure of a software library

The specifics for the libraries used are mentioned above. To implement these libraries into a sufficient structured library there are different insight, tactics and strategies for setting up such a library.

As can be read in the previous parts of this chapter most of the libraries are already made, documented and ready to use for most programmers. As one wise teacher once told a class full of students: “A good programmer has a certain amount of laziness, to not make code that has already been made”[10]. When looking specifically at this context, the python library, there are a few options that can be executes concerning the library structure.

Making a wrapper library
Making a wrapper library is similar to the solution used for the C++ library, by using libraries that are already made but combining them in one library. While the wrapper allows for a universal library structure it costs a more time and has a certain amount of overload that is not as efficient.

Making demos with the current libraries
Since most libraries offer their own tutorials and setups the only thing that needs to be explained is how the different libraries work with each other. This would require a clear manual, tutorials and easy access to the documentation of the other libraries. The amount of the time used is reduced since the entire wrapper doesn’t have to be made.

4.2.2 Usability

Before deciding what structure is best to use for the python software first the effects of different choices on the overall usability of the software. Unlike with the CE students there is no universal library that has been used by all students. Because this is not the case increasing the usability of the software can be hard to achieve.

When learning a new programming language or just a new library the focus in classes are on examples of the code, practicing with exercises and explanations from the teacher. The last two require a teacher for feedback and/or explanations, since the makers of the libraries will not be present for this project and the use of this platform this is a less than ideal option. Because of this limitation the documentation and demos of the libraries become even more important. For students

[7] https://picamera.readthedocs.io/en/release-1.13/
[8] https://learn.adafruit.com/capacitive-touch-sensors-on-the-raspberry-pi/programming
[9] https://www.modmypi.com/blog/hc-sr04-ultrasonic-range-sensor-on-the-raspberry-pi
[10] Van Ooijen, W. (September 2015)

(and other people) to learn a new library the code has to be well documented and examples need to be available to increase usability.

4.2.3 Maintainability

As mentioned before most of the libraries available for the hardware are implemented by other people, it is yet to be determined if this has any effect on the maintainability. For example, change in one library could cause loss of functionality in the other libraries. This can certainly be a problem but the commitment to updating this flaw will be bigger with most of the libraries that are not made by a student that is likely to graduate within a year.

4.3 Conclusion

The choice mainly comes down to making a wrapper or making support for the current libraries. Both choices have advantages. The wrapper allows for a consistent library structure that most of the libraries are lacking when combined. While making support for the combined usage of the libraries takes less time.

As mentioned above the usability increases by documenting the library and having examples for people to try. The maintainability is increased by having regular updates for the library.

Making a wrapper doesn’t necessarily increases the usability since it more about the documentation that must be made to go along with the library. Creating support does increase the usability.

Concerning the maintainability of the software a creator that is committed to fixing bugs is key. While there is no indication that the creators of the libraries mentioned above will do this there is however the situation where the creator of the wrapper (Jip Galema) is not so committed to updating the software. Because of these two factors concerning the usability and maintainability the choice for the python software will be adding support to the existing libraries by making a wiki and examples on GitHub (where both language software will be situated).

5. Conclusion

As can be read in the assignment chapter the main question of the research is:

> *What is the driving platform blueprint that featured usability in current second-year project, for the bachelors Computer Engineering and Applied Artificial Intelligence, and is maintainable for the next years and other projects?*

While part of this research was not completed, this would be the product review, part of this question can be answered. Before doing this, it is important to go through all the different decisions made within the hardware architecture and software architecture.

Hardware architecture
To increase the usability and maintainability of the platform the following components will be used:
- Raspberry Pi 3 Model B+
- JGA25-370 DC gearbox with encoder, 6V – 900 mA
- L298N H-brigde
- 2 MT3608 DC-DC Booster step up 2V-24V to 5V-28V, 2A
- Lithium battery 3.7V 8000 mAh.
- 4 3D printed wheels
- Metal axle bar
- Wooden platform
- 2 3D printed motor holders
- Screws and bolts

The decisions mounting to this component range from increasing learnability, for example choosing the Raspberry Pi over a lesser known pine64 to straight up price like with the DC-motors.

Software architecture
When making decisions for the hardware architecture one of the decisions made for the software architecture was the languages, since CE-students program almost entirely in C++ and AAI-students have worked with python. To increase the usability of the software it was decided to add Raspberry Pi support to the hwlib library, a hardware library used in most programming classes of CE-students. For the python software the decision was made to combine different libraries and create documentation and examples of how to use these together.

Conclusion
Knowing this the part of the research that can be answered is a platform blueprint that is not necessarily approved by the clients but does take their requirements into consideration. This conclusion would be:

The driving platform blueprint that features both usability in current second-year projects, for the bachelors Computer Engineering and Applied Artificial Intelligence, and is maintainable for next years and other projects would be a platform that has the following hardware components:
- Raspberry Pi 3 Model B+
- JGA25-370 DC gearbox with encoder, 6V – 900 mA
- L298N H-brigde
- 2 MT3608 DC-DC Booster step up 2V-24V to 5V-28V, 2A

- Lithium battery 3.7V 8000 mAh.
- 4 3D printed wheels
- Metal axle bar
- Wooden platform
- 2 3D printed motor holders
- Screws and bolts

And has both C++ and Python software support with documentation and examples to decrease the time it takes for students and other people within school to learn the different software features.

6. Recommendation

As can be read in the assignment chapter this research didn't go as planned. This is mainly due to the lack of time and some hardware that didn't work according to plan. There are a couple of recommendation that can be made to increase the effectiveness of the platform that where beyond this research control.

Voltage boosters
As told before the hardware didn't exactly react the way it was described an expected, this brought with it a lot of difficulties. In further research designing a booster specific for this platform that does meet the requirements is probably the best option. This will fix a lot of the problems encountered when searching for a power source.

Overall testing
Due to the lack of time the platform consistent with this blueprint was never built. This didn't allow for decisions to be made on testing results and bring to light other problems that would probably arise from building the platform. Because of this the research should mainly be used as a theoretical blueprint since it cannot be guaranteed this will work.

7. Sources

(PEng), K. M. (2015, 9 14). *DC Motors vs. Stepper Motors for Motion Control Applications*. Retrieved from ECN: https://www.ecnmag.com/blog/2015/09/dc-motors-vs-stepper-motors-motion-control-applications

Atmel. (n.d.). *SAM3X / SAM3A Series.* Retrieved from Amtel: http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-11057-32-bit-Cortex-M3-Microcontroller-SAM3X-SAM3A_Datasheet.pdf

Cambridge Business English Dictionary © Cambridge University Press. (n.d.). *Usability*. Retrieved from Cambridge dictionary: https://dictionary.cambridge.org/dictionary/english/usability

Cawley, C. (2015, May 28). *Pi to Go? 3 Ways of Powering a Raspberry Pi for Portable Pi for Portable Projects*. Retrieved from makeuseof: https://www.makeuseof.com/tag/pi-go-x-ways-powering-raspberry-pi-portable-projects/

ElectronicsTutorials. (n.d.). *DC Motors*. Retrieved from Electronics Tutorials: https://www.electronics-tutorials.ws/io/io_7.html

ELPROCUS. (n.d.). *What Is The Difference Between DC Motor, Servo Motor And Stepper Motor*. Retrieved from ELPROCUS: https://www.elprocus.com/difference-dc-motor-servo-motor-stepper-motor/

Gaugette. (2013, January 14). *Rotary Encoder Library for the Raspberry Pi*. Retrieved from Gaugette: https://guy.carpenter.id.au/gaugette/2013/01/14/rotary-encoder-library-for-the-raspberry-pi/

Group, L. H. (2014, October 23). *What is Software Reuse?* Retrieved from https://web.archive.org/web/20141023013122/http://lombardhill.com/what_reuse.htm

GTSparkplugs. (n.d.). *Dropping Resistor Calculator*. Retrieved from GTSparkplugs: http://www.gtsparkplugs.com/Dropping_Resistor_Calc.html

ISO25000. (n.d.). *ISO/IEC25010*. Retrieved from ISO25000 software product quality: https://iso25000.com/index.php/en/iso-25000-standards/iso-25010

Khan, S. (2012). *Khan Academy*. Retrieved from Resistors in Series: https://www.khanacademy.org/science/physics/circuits-topic/circuits-resistance/v/circuits-part-2

Kiwi electronics. (n.d.). *Raspberry Pi 3 Model B+*. Retrieved from Kiwi electronics: https://www.kiwi-electronics.nl/raspberry-pi-3-model-b-plus?src=raspberrypi

lefticus. (2014, April 28). *C++ Coding Standards*. Retrieved from GitHubGist: https://gist.github.com/lefticus/10191322/revisions

MiniInTheBox. (n.d.). *18650 Battery 5000 mAh 4pcs Rechargable for Camping / Hiking / Caving*. Retrieved from MiniInTheBox.com: https://www.miniinthebox.com/en/p/5000mah-18650-battery-4pcs_p1490235.html?currency=EUR&litb_from=paid_adwords_shopping&country_code=nl&utm_source=google_shopping&utm_medium=cpc&adword_mt=&adword_ct=202532173142&adword_kw=&adword_pos=1o3&adword_pl=&adword

ModMyPi LTD. (2013, Augustus 25). *What's The Difference Between DC, Servo & Stepper Motors?* Retrieved from MODMYPI: https://www.modmypi.com/blog/whats-the-difference-between-dc-servo-stepper-motors

NVIDIA. (n.d.). *Jetson TK1*. Retrieved from NVIDIA: https://www.nvidia.com/object/jetson-tk1-embedded-dev-kit.html

Open Impulse. (n.d.). *JGA25-370 DC Gearmotor (176 RPM at 6V)*. Retrieved from Open Impulse: https://www.openimpulse.com/blog/products-page/25d-gearmotors/jga25-370-dc-gearmotor-176-rpm-6-v/

Pololu. (n.d.). *4.1 Choosng the power supply, Tic and stepper motor*. Retrieved from Pololu: https://www.pololu.com/docs/0J71/4.1

Raspberry Pi foundation. (n.d.). *Homepage*. Retrieved from Raspberry Pi: https://www.raspberrypi.org/

Roger's HobbyCenter. (n.d.). *A Guide to Understanding LiPo Batteries*. Retrieved from Roger's HobbyCenter: https://rogershobbycenter.com/lipoguide/

TranThinh. (n.d.). *Motor With Encoder, How to Read Input Vaue From Encoder*. Retrieved from Instructables: https://www.instructables.com/id/Motor-With-Encoder-How-to-Read-Input-Value-From-En/

Unknown. (n.d.). *Arduino (computerplatform)*. Retrieved from Wikipedia: https://nl.wikipedia.org/wiki/Arduino_(computerplatform)

Unknown. (n.d.). *DC Motor*. Retrieved from Wikipedia: https://en.wikipedia.org/wiki/DC_motor

Unknown. (n.d.). *ESP8266*. Retrieved from Wikipedia: https://en.wikipedia.org/wiki/ESP8266

Unknown. (n.d.). *I2C*. Retrieved from Wikipedia: https://en.wikipedia.org/wiki/I%C2%B2C

Unknown. (n.d.). *Serial Peripheral Interface* . Retrieved from Wikipedia: https://en.wikipedia.org/wiki/Serial_Peripheral_Interface

8. Appendix

8.1 Assignment plan

1. Introduction

As a student that likes programming one of the most difficult thing about developing on new hardware and software (for me at least) is getting started. Just like many of my fellow programmers we rely a lot on datasheet to get information from chips and documentation for software libraries. When either one is missing the amount of work it cost to learn something new is exponentially increased. One of my biggest traumas in this department was the inspiration for this research.

In a fast growing and changing industry like hardware and software development, usability and maintainability are key to save time on projects. Especially in a hardware industry where every year there are new processors with new instruction sets, that works differently than before. As student with not as much knowledge and experience on the specific chips the usability of libraries that are already well known and documented are often a lifeline within a project.

Students that are in the beginning phases of learning how to program are often dependent on the usability and maintainability of libraries. Libraries that don't feature examples are often last on the list. Therefore, whenever there is little usability or maintainability in libraries that have to be used progress can be delayed, and a lot of suffering will happen.

This research plan is about setting up a driving platform that can be used by second year students of both Computer Engineering and Applied Artificial Intelligence bachelors that is both usable and maintainable. As part of the research there will be an extended look into the characteristics that make both a driving platform and its code usable and maintainable.

1.1 The University of Applied Sciences Utrecht

This research will be part of the bachelor Computer Engineering and the bachelor Applied Artificial Intelligence. The platform will be mainly used within the R2D2 (Computer Engineering) and BB-8(Applied Artificial Intelligence) projects. These projects consist of the whole second year worth of students from a bachelor. The students will be divided into different groups that then develop different modules, using project methods like scrum.
The three clients are Huib Aldewereld, Wouter van Ooijen and Arno Kamphuis. They will set requirements on the needs of both departments. Huib Aldewereld and Arno Kamphuis are in the curriculum commission of the bachelor Applied Artificial Intelligence while Wouter van Ooijen is chairman of the curriculum commission of the bachelor Computer Engineering.

1.2 The context

The R2D2-project makes it look like the students are setting up a company. This company is making hardware modules. Because the group of students is often quite large (about 30-40 students) this is also a good time to learn more about different developing techniques, like scrum. As part of this project the students will brainstorm different hardware modules that can be made with the available hardware. The requirements for these modules are set. After this step the group is split into smaller groups of about 5 students, called project groups. As part of the hierarchy within the 'company' the scrum masters will come together every week to explain what has been accomplished. After a week every group takes on a new project, this can either be: setting up something new or building extra features on top of a project from another group.

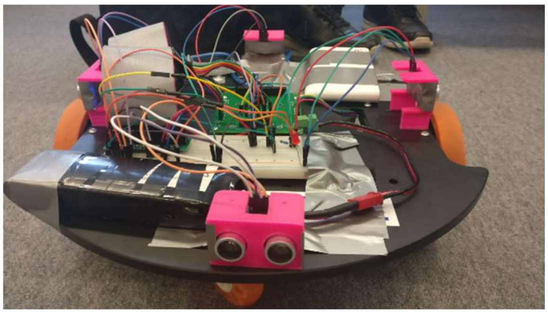


Figure 1.1 The current state of the rosbee

As a student I've completed the R2D2 project, this gave me some insight on what the it's like to work with a lot of people but also showed some of the problems with the current hardware platform that is being used, a platform called the rosbee (see figure 1.1). In previous years the rosbee has left a lot to be desired. The fact that the platform cannot be found on the internet (this is the first photo that came up which was from the R2D2-project two years ago) and the code hasn't been updated for 3 years is really telling about how up to date the platform is. That makes the platform not ideal to use in this situation. As explained in the introduction students that are depended on platforms that are easy to use and maintainable, especially in the early years of the bachelor. The rosbee only features one example and has almost no documentation on platform specific parts.
In previous experienced with the R2D2 project by students and myself this platform was not build for adding sensors or supporting the microcontroller that was used in the project, this took 2 weeks to fix. This is quite a lot of time when a project lasts 6 weeks.

This year is the first year of the bachelor Applied Artificial Intelligence. This means that the bachelor Computer Engineering will be focused on developing firmware while the bachelor Applied Artificial Intelligence is about developing algorithms.
Because the rosbee is mostly a hardware project these students will not be doing anything with programming the hardware directly. Although algorithms for driving platforms can be a challenging and suitable module for the BB-8 project. For this to be possible a new platform must be designed and built that is usable and maintainable for both bachelors.

2. The assignment

The goal of the assignment, as explained in the introduction, is to design and build a driving platform that can be used for educational purposes within the bachelors Applied Artificial Intelligence and Computer Engineering. The most important thing is that the platform is usable first, and maintainable second.

2.1 Phases of the project

As part of the assignment there will be three phases: Analysis, design and realisation.
The analysis phase will be focused on the hardware/software that increase the usability and maintainability of the driving platform and the decision that must be made to get to an architecture. This architecture will consist of the different hardware parts that go into the platform and their arguments.
The second phase is the design phase, this phase will focus on researching the software part of the platform. After the architecture is set there will be a lot of software that has to be made. The requirements of the software will be somewhat vague, for example: make sure the microcontroller can communicate with the motors. These requirements will be finetuned in this phase. As part of this phase there will be research into making these hardware/software libraries usable and maintainable. This phase will also be used to test this code on the hardware. At the end of the phase there will be a blueprint. This features the architecture as well as the structure of the software and the working of it. The blueprint will feature class diagrams and STD's to give a clear view of the project architecture
The last phase is the realisation phase. After the first two phases there is an architecture and a blueprint. From this architecture a big part of the code has to be made and tested. The only thing left to do is to discuss with the clients if the requirements are met and build the platform. This will be mainly a practical phase and there is very little opportunity or need for more in-depth research. At the end of the phase there will be a platform and a manual for constructing the platform and a documented hardware/software library.

The following items must be delivered:

- Platform
- Documented code library
- Research plan
- Research
- Architecture document (list of hardware and software components)
- Assembly manual

2.2 Project method

This project will use the agile method. This is ideal since there needs to be a lot of feedback from the clients in how the platform is going to look hardware and software wise.

2.3 Main question and sub question

To make this driving platform there needs to be some research on the hardware/software components. To understand the terms used in the sub-question there will be a glossary below:

Architecture: A document of components that are used in the driving platform with the arguments to use these components
Blueprint: A document with tested components and a thought-out structure for the code, as well as an system architecture. The only thing that has to happen after this document is connect the different code modules and assembling the platform.

The main research question will be:

> What is the driving platform blueprint that featured usability in current second-year project, for the bachelors Computer Engineering and Applied Artificial Intelligence, and maintainable for the next years and other projects?

To answer the main research question there must be sub-questions, these are below:
- What is an architecture for a driving platform that is both usable and maintainable?
- What is a blueprint for a hardware and software library that is both usable and maintainable within the context of the driving platform?
- What aspects of the blueprint live up to the standards of the clients, and what aspects don’t?

3. Theoretical Framework

When looking at the main and sub-questions there are three concepts that will be central to this research: Usability, maintainability and the driving platform.

3.1 Usability

The meaning of the term usability within the software development standard ISO25010 consist of the following definition: “Degree to which a product or system can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use.” (ISO25000, n.d.)

As part of this definition the following sub-characteristics are defined:

- **Appropriateness recognizability:** Degree to which users can recognize whether a product or system is appropriate for their needs.
- **Learnability:** Degree to which a product or system can be used by specified users to achieve specified goals of learning to use the product or system with effectiveness, efficiency, freedom from risk and satisfaction in a specified context of use.
- **Operability;** Degree to which a product or system has attributes that make it easy to operate and control.
- **User error protection:** Degree to which a system protects users against making errors.
- **User interface aesthetics:** Degree to which a user interface enables pleasing and satisfying interaction for the user.
- **Accessibility:** Degree to which a product or system can be used by people with the widest range of characteristics and capabilities to achieve a specified goal in a specified context of use

When adjusting the usability definition to the specific context it gives a better picture of what the concept should accomplish, the specific definition for usability within the project would be:

The degree to which the platform (and its software) can be used by students to make hardware/software modules with effectiveness, efficiency and satisfaction within the R2D2 and BB8 projects.

The main focus of this research will be on enhancing the learnability and operability, these two characteristics are within the scope of the project.

3.2 Maintainability

The definition of the term maintainability within the software development standard ISO25010 is: “The degree of effectiveness and efficiency with which a product or system can be modified to improve it, correct it or adapt it to changes in environment, and in requirements.” (ISO25000, n.d.)

This definition can be divided into 5 sub-characteristics:

- **Modularity.** Degree to which a system or computer program is composed of discrete components such that a change to one component has minimal impact on other components.

- **Reusability.** Degree to which an asset can be used in more than one system, or in building other assets.
- **Analysability.** Degree of effectiveness and efficiency with which it is possible to assess the impact on a product or system of an intended change to one or more of its parts, or to diagnose a product for deficiencies or causes of failures, or to identify parts to be modified.
- **Modifiability.** Degree to which a product or system can be effectively and efficiently modified without introducing defects or degrading existing product quality.
- **Testability.** Degree of effectiveness and efficiency with which test criteria can be established for a system, product or component and tests can be performed to determine whether those criteria have been met.

When adjusting the maintainability definition to the specific context it gives a better picture of what the concept should accomplish, the specific definition for usability within the project would be:

The degree of effectiveness and efficiency with which the platform and software can be modified to improve it, correct it or adapt it to changes in environment, and in requirements.

The main focus to enhance the maintainability will be on modularity and reusability, these two characteristics are within the scope of the project.

As part of this research the different decisions that come with increasing the degree of the sub-characteristics are reasoned and documented, this will give a clear view of the type of platform that needs to be build and how the software needs to be structured. Usability will always come first since after all this is still a project for the near future.

3.3 Driving platform

The driving platform is the concept that brings both concepts mentioned above together. The driving platform will have motors and wheels and a platform to connect and place electronics on, and while it might not be as important as the other two concepts it is still where most of the usability and maintainability is based on.

The goal of the platform is to meet the requirements set in part of the MoSCow analysis on page 10.

3.4 Important literature

A big part of this research will be looking through papers that have tackled similar problems. Papers can be found on different databanks like:

- Google Scholar
- Web of Knowledge

Some of the relevant search terms that can be used in these databanks are: driving platform, reusable hardware library, usability in software. Other terms that are relevant to the research and can be used are: platform, hardware, reusability, maintainability, robot, ISO25010

The other part of the research is working through the code of other libraries that are being maintained throughout the years to find different characteristics that make up such a library.

3.5 Significance of this research

The importance of the project can be traced back to the fact that teachers found it to be useful and relevant, otherwise this project wouldn't have been made. In the greater scheme of the overall education system it is worth noting that the research (when done right) will give great insight into the choices that have to be made when designing a platform.

As part of my personal development the research is relevant as I've never really made a project that can be used by other students.
In scheme of the university it is important to set up the driving platform that can be used for years to come. For this driving platform to be successful the base and initial code is crucial.

3.6 Research method

The main project phases that are of any relevance in the project are: Analyses, Design and Realisation. For the analysis there will be a literature study, this prioritize a structured approach and no initial bias, and will be about the different characteristic of a usable and maintainable platform and its software, and what already has been used. The design phase will be focused on a IT architecture sketching to design the system and a product review with the clients to validate the results found in the analysis phase and choose a blueprint based on these results. Finally, the realisation phase will be used to build the architecture as a prototype.

Sub-question	Method	Result
What is an Architecture for a driving platform that is both usable and maintainable?	Literature study	A document with the decisions on what hardware to use and the arguments on why to use them
What is a blueprint for a hardware and software library that is both usable and maintainable within the context of the driving platform?	Literature study and IT architecture sketching	A document with the recommended code structure and tools that can be used (and the argument for it), and class diagrams and other diagrams that give a clear view of the structure
What aspects of the blueprint live up to the standards of the clients, and what aspects don't?	Product review	A review by the clients that can be used as a feedback moment.

4. Quality and Planning

As explained in the theoretical framework the quality of the product will be measured by applying the ISO25010 software development standard. As part of this standard the priority will always go to the usability of the project whenever something makes the platform more maintainable but less usable.

4.1 MoSCoW-analysis

To get a better view of the requirements that will be used throughout the research for the platform, a MoSCoW-analysis has been made:

The must haves:

- The platform must be able to turn
- The platform must be able to drive forward
- The platform must have enough room to connect different sensors to it
- The platform must have implementation that can be used by students from both Computer Engineering and Artificial Intelligence

The should haves:

- The platform should be able to turn a given angle with an offset accuracy of 5 degrees
- The platform must be able to drive straight for a given amount of distance and an accuracy of 10 cm.
- The platform should have an extension that can be used to add surface area for extra sensors.
- The platform should be easy to reproduce whenever there is an extra one needed
- The platform should have an easy way to change parts and sensors
- The platform should have sensor that support autonomous driving project (a camera or an ultrasonic sensor)

The could haves:

- The platform could have support for remote control
- The platform could have support for more than two microcontrollers.

The won't haves:

- The platform will not have a mobile app with Bluetooth connection
- The platform won't have an autonomous driving algorithm

4.2 Standards, methods and tools

During the project a few standards, methods and tools will be used. For the code the C++ coding standard from lefticus will be used.[11] While there may be slight changes to support the embedded side of this project the base will always be this standard.
The code will be stored on GitHub and will be commented with Doxygen. The editor for the code will be Visual Studio Code.

[11] https://gist.github.com/lefticus/10191322

To encourage structured working trello will be used as a scrum tool. The board will feature backlog items and deadlines.

4.3 The planning

This project is structured in a way that research needs to be done first. This way making a prototype is easier and more efficient. The initial deadlines are below.
Assignment plan deadline: 9 October 2018
Further research so the correct parts can be ordered: 6 December 2018
Architecture document deadline: 8 November 2018
Blueprint document (with product-review) deadline: 17 December 2018
Realisation phase deadline: 21 January 2018
Overall deadline: 1 February 2018

4.4 Risks

The scope of the project is too big
The project is divided into three phases (analysis, design and realisation) the scope of the project could be too big.
Completing all three phases is the goal but is not necessary for the project to be valuable. When in one way or another the scope is too big, realisation will be the first phase not completed in the project. While this is unfortunate, there can still be another student that can finish this project by assembling the platform. The project has to be structured in a way that every phase can be used by other students and researchers for either the same or different purposes.

The hardware can't be delivered on time
While the idea of the 'parts' deadline is to prevent this type of problem, with the uncertainty of both delivery times of AliExpress and not knowing the exact parts this can still happen when things are ordered later then the deadline. At the end of the design phase the parts for the driving platform should be known and can still arrive on time. Whenever there isn't a part on time the platform can be assembled until that part is essential. Whenever the part arrives a student can assemble the rest of the platform. As part of the maintainability it would be a good test to see if the instructions are clear enough for someone to assemble the platform.

5. Sources

Cambridge Business English Dictionary © Cambridge University Press. (n.d.). *Usability*. Retrieved from Cambridge dictionary: https://dictionary.cambridge.org/dictionary/english/usability

Group, L. H. (2014, October 23). *What is Software Reuse?* Retrieved from https://web.archive.org/web/20141023013122/http://lombardhill.com/what_reuse.htm

ISO25000. (n.d.). *ISO/IEC25010*. Retrieved from ISO25000 software product quality: https://iso25000.com/index.php/en/iso-25000-standards/iso-25010

lefticus. (2014, April 28). *C++ Coding Standards*. Retrieved from GitHubGist: https://gist.github.com/lefticus/10191322/revisions

6. Appendix

1. Quality and requirements for assignment plan

Object	Criterium	Indicatoren	
Creatief Probleem Oplossen			
Opzet en beschrijving	Verstrekte opdracht	Feitelijk	• Weergave van de opdracht zoals oorspronkelijk door de organisatie is aangedragen.
	Aanleiding van de kwestie	Kernachtig	• Gebeurtenissen die aanleiding zijn geweest voor de opdrachtgever om een opdracht te formuleren.

	Kwestie	Kernachtig	• Beschrijving van de kwestie door de ogen van de student. Dit is bij een aanvraag nog summier en kan verschillen van de opdracht zoals verstrekt door de opdrachtgever:
			  ○ Wat is de kwestie? Voor wie is het een kwestie?
			  ○ Is het een probleem of een kans?
			    ▪ Bij een probleem: Waaraan zie je dat het een probleem is? Wat zijn mogelijke oorzaken van het probleem? Wat zijn de gevolgen van het probleem?
			    ▪ Bij een kans: Wat wil de

			organisatie met de kans bereiken? Hoe wil de organisatie de kans bereiken? Wat is daarvoor nodig?
	Doelstelling	Helder	Wat is: • de doelstelling/missie voor de opdrachtgever. Waarom pakt hij/zij de kwestie op? • de doelstelling van deze opdracht. Welk deel van de doelstelling van de opdrachtgever moet opgelost zijn om de opdracht geslaagd te noemen?
		Richtinggevend	• De oplossingsruimte is beschreven.
		Afgebakend	Het soort opdracht is duidelijk. 1. Descriptief onderzoek: het antwoord op een descriptieve vraag met eventueel een aanbeveling hoe iets verbeterd kan worden. 2. Advies/Ontwerp: een onderbouwd en uitgewerkt advies hoe iets verbeterd kan worden en/of ontwerp voor een systeem, apparaat, software etc. eventueel met implementatieplan. 3. Product/Proof of concept: ontwerp en realisatie van een systeem, product, software.
	Eindresultaat	Specifiek	• (De aard van) het gewenste resultaat/product van deze opdracht is helder en specifiek beschreven.

	Globale aanpak en methoden	In beeld	• Er is in grote lijnen zicht op de aanpak en de daarbij te gebruiken methoden om tot het eindresultaat te komen.
	Concept hoofdvraag of opdracht in enkele zinnen	Aanwezig	• Concept-versie in enkele consistente samenvattende zinnen die zicht geven op de kwestie, het gewenste eindresultaat, de centrale begrippen en de te gebruiken methode (soort opdracht).
Aard, niveau en haalbaarheid	Hoofdvraag/Opdracht	Relevant	• Probleem is nog niet opgelost voor de opdrachtgever. • Beoogde oplossing is de moeite waard in de probleemcontext. • (Beoogde oplossing is maatschappelijk/praktisch de moeite waard).
		Precies	• Gewenste resultaat is specifiek te beschrijven. De oplossing zit niet al in de doelstelling ingebouwd.

		Afgebakend op niveau van de HBO-ICT
		
		afstudeerrichting
			• Complexiteit: minimaal 2 ICT-beroepstaken op niveau 3 (in één architectuurlaag) van de HBO-ICT afstudeerrichting (zie studiegids 2015-2016 H.2.1.3);
			• Zelfstandigheid professional skills:
			  ○ Creatief probleem oplossen én Analyse en informatie-verwerking én Planning & organisatie: Situatiegericht;
			  ○ de overige minimaal: Probleemgericht.

			• Er is nog ruimte om op zoek te gaan naar de oorzaak van het probleem en de verstrekte opdracht daarop bij te stellen. • Realiseren van de doelstelling vereist onderzoek door de student waarin verschillende opties worden onderzocht en afgewogen. • Bij eventuele geheimhouding moeten de examinatoren wel alle relevante stukken in kunnen zien.
		Functioneel	Het soort opdracht is helder beschreven als: • descriptief onderzoek óf • advies/ontwerp óf • product/proof of concept.
Onderzoek Doen			
Opzet en beschrijving	Kennisdomein in relatie tot de ICT-beroepstaken	Helder	• Beschreven wordt bij welk kennisdomein de vraag aansluit. • Beschreven wordt in welk kennisdomein de oplossing gezocht wordt. • Op basis hiervan moet (door de beoordelaar) de doelstelling van de opdracht kunnen worden vertaald naar de ICT-beroepstaken en het niveau daarvan.
	Voorlopige literatuurlijst	Aanwezig	• Literatuur die de student verwacht te moeten bestuderen en gedeeltelijk al doorgenomen heeft. • APA-notatie.
Aard, niveau en haalbaarheid	Hoofdvraag/Opdracht	Inhoudelijk verankerd binnen afstudeerrichting	• Kennis ICT-beroepstaken valt binnen de afstudeerrichting.
Leiderschap, Samenwerken en Communicatie			
Opzet en beschrijving	Actoren	Helder	• Functie en opleidingsniveau bedrijfsbegeleider (indien

			mogelijk met verwijzing naar LinkedIn). • Andere betrokkenen en hun benaderbaarheid.
	Communicatieve verzorging	Inhoud goed gecommuniceerd	• Objectief, precies, zorgvuldig, helder, beknopt, zakelijk, correct taalgebruik (zie de criteria bij communicatieve verzorging voor een toelichting op deze indicatoren).
		Volledig	• Voldoet aan inhoudsrichtlijnen in afstudeerleidraad (zie inhoud afstudeervoorstel).
		Verzorgd	• Voldoet aan regels voor: spelling en interpunctie, redactionele vormgeving, literatuurlijst en bronvermelding.
Aard, niveau en haalbaarheid	Actoren	In beeld en benaderbaar voor student	• Aanwezigheid bedrijfsbegeleider. • Andere betrokkenen.
		Opleidingsniveau	• Bedrijfsbegeleider heeft minimaal HBO-niveau.
Planning, Organisatie, Kwaliteit en Ethiek			
Opzet en beschrijving	Organisatorische context van de kwestie	Kernachtig	• Korte beschrijving van soort, omvang en structuur van de organisatie die de opdracht heeft gegeven. • Korte beschrijving van het onderdeel waarvan de opdracht afkomstig is. • Plaats van de student in de organisatie.
	Planning	Kernachtig	• Afspraken over looptijd van de opdracht, inzet van uren en ruimte voor terugkomdagen en rapportages aan school.
	Ethische afweging van de doelstelling	Helder	• In het voorstel wordt aandacht besteed aan relevante maatschappelijke, normatieve en ethische

			aspecten waaronder duurzaamheid.
	Risico's	Kernachtig	• Beschrijving van belangrijkste risico('s) voor het slagen van het project naast ziekte/uitvallen student of opdrachtgever.
Aard, niveau en haalbaarheid	Planning	Haalbaar	• Beantwoorden van de vraag moet voldoende werk opleveren gedurende de onderzoeksperiode, rekening houdend met de beschikbaarheid van informatie en het aantal betrokkenen, afdelingen en processen. • Beantwoorden van de vraag moet haalbaar zijn in de onderzoeksperiode, gegeven de hiervoor genoemde factoren.
	Vraag/Opdracht	Ethisch afgewogen	• De opdrachtgever biedt een omgeving waarin ruimte is voor een afweging van relevante maatschappelijke, normatieve en ethische aspecten waaronder duurzaamheid.
	Risico	Hanteerbaar	• De risico's zijn van dien aard dat de afstudeeropdracht met hooguit één periode uitstel haalbaar is.
Leren en Persoonlijke Ontwikkeling			
Opzet en beschrijving	Persoonlijke ontwikkeling	Helder	• De student beschrijft helder en overtuigend waarin de persoonlijke uitdaging op het gebied professional skills ligt (creatief probleem oplossen, Onderzoek doen, leiderschap, samenwerken, communiceren, plannen

			en organiseren en/of ethisch handelen). (Op basis hiervan en de doelstelling moet de beoordelaar kunnen inschatten wat het niveau van de professional skills voor de opdracht is).
Aard, niveau en haalbaarheid	Persoonlijke ontwikkeling	Voldoende uitdaging	• Opdracht sluit aan bij persoonlijke ontwikkeling.

8.2 Test report voltage

TEST REPORT: VOLTAGE BOOSTERS

DATE

11-12-2018

AUTHOR

JIP GALEMA (1679915)

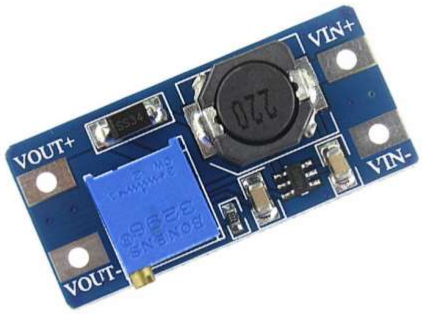
ORGANISATION

UNIVERSITY OF APPLIED SCIENCES UTRECHT

VERSION 1.0

TEST REPORT: MT3608 DC-DC BOOSTER STEP UP 2V-24V TO 5V-28V

Hardware specifications:
Incoming voltage: 2-24V
Outgoing voltage: 5-28V
Outgoing current (max): 2A
Conversion-efficiency: 96 %



Test case
Because this voltage booster will be used with a single cell LiPo battery the testing voltage will be 3 volts. This is mainly because the 3.7 V per cell is an average and 3 volts is the worst-case scenario (an almost empty battery). The current will be 2A

The testcase will be:
- Voltage with input of 3V and desired output voltage of 5 and 6V.
- Current with input of 3V and desired output voltage of 5 and 6V.

Results
1. Voltage with input of 3V and desired output voltage of 5 and 6V.
   The voltage is stable in both situations and is easy to adjust for either 5 or 6V.
2. Current with input of 3V and desired output voltage of 5 and 6V
   The current at 5V is 700 mA, the current at 6V the current is 600 mA. The booster gets really hot in both cases.

Conclusion
So while the MT3608 voltage works really well and is consistent the current in this situation is quite low.