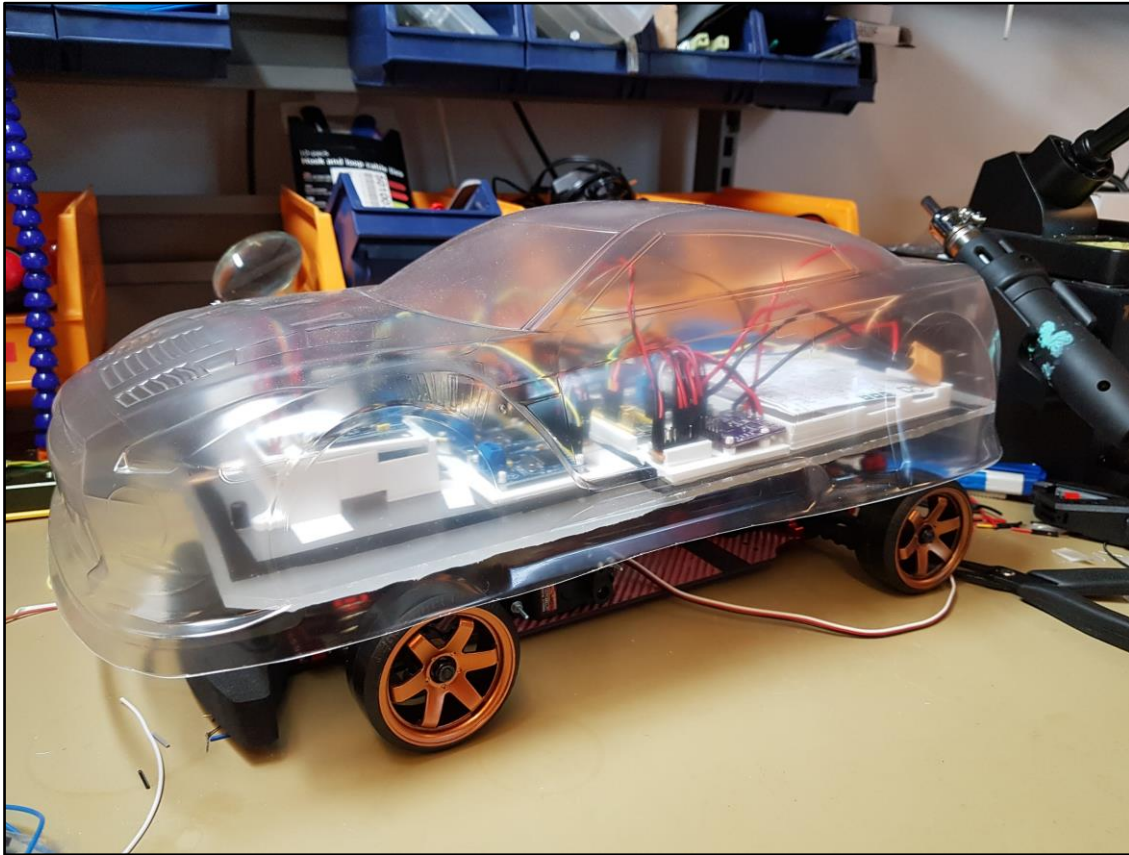




CHALMERS



SCALABILITY OF ELECTRIC DRIVETRAINS

Bachelor's thesis in mechatronics engineering

Simon Hellemar & Jesper Mass

ACKNOWLEDGEMENTS

For supervising our bachelor's thesis, we would like to thank Bertil Thomas at Chalmers University of Technology.

We also want to thank Pierre Ekwall and Fredrik Lundgren at Infotiv AB for giving us the opportunity to do our thesis work at Infotiv and guiding us along the way.

We would also like to thank Joachim Lublin and Olle Norelius at Infotiv AB for their great technical support and equally great company during our work at Infotiv.

Simon Hellemar, Jesper Mass, Gothenburg, May 2018

ABSTRACT

This research covers the design of electric drivetrains and what differences there are between different sizes of motors and their allocated control circuits. The main type of motor covered in this report is the brushless direct current motor or BLDC motor, but other common motor types are briefly explained as well.

During the project an electric drivetrain circuit board was designed for a radio-controlled car to be used for education at Infotiv AB. The car is equipped with a BLDC motor rated for 11.1 volts and 1.3 ampere, a steering servo and a network of modules that emulate the inner workings of a real car.

The conclusion of this project is that not too many components need to be exchanged to accommodate for higher voltages. Mainly the inverter and gate driver for the inverter needs to be upgraded. When driving motors that draw a lot of current, active cooling will be needed to avoid overheating of the circuit. Depending on what gate driver is used, even though PWM is used to control, the other pins on the driver might differ in purpose. This will affect parts of the code and if more pins are needed there are versions of the MCU in the same series with even more pins but still work the same.

TABLE OF CONTENTS

Terminology/ Abbreviations	1
1 INTRODUCTION	2
1.1 Background	2
1.2 Purpose.....	2
1.3 Scope.....	2
1.4 Specifications of the project.....	3
2. THEORY	4
2.1 Brushed DC motors.....	4
2.2 BLDC motors.....	4
2.3 Induction AC motors.....	5
2.4 Eagle	5
3. METHOD	7
3.1 Procedure	7
3.1.1 Pilot studies.....	7
3.1.2 Electronics design	8
3.1.3 PCB design.....	10
3.1.4 Programming of MCU.....	11
3.1.5 Testing.....	11
3.2. Research.....	11
4. DESIGN	13
4.1 Designing the PCB.....	13
4.2 Solution alternatives.....	16
5. RESULTS	17
5.1 Motors and power	17
5.2 Transistors and drivers	17
5.3 Control	17
5.4 Setup in varying scales.....	19
6. CONCLUSIONS/ DISCUSSION	20
REFERENCES	21

Terminology/ Abbreviations

A - Ampere

AC - Alternating current

BLDC - Brushless DC motor

BOM - Bill of Materials

CAD - Computer Aided Design

CAN - Controller area network

DC - Direct current

EMC - Electromagnetic compatibility

EMF - Electromotive Force

ESC - Electronic speed controller

IC - Integrated circuit

MCU - Microcontroller unit

MOSFET - Metal oxide semiconductor field effect transistor

PCB - Printed circuit board

PMAC - Permanent magnet AC motor

PWM - Pulse width modulation

SPI - Serial peripheral interface

SWD - Serial wire debug

UART - Universal asynchronous receiver transmitter

VCU - Vehicle control unit

V - Volt

W - Watt

1 INTRODUCTION

1.1 Background

As the world shifts more towards a sustainable future, interests in electric vehicles are increasing. But the reduced CO₂ emissions are not the only reasons for using electric motors. Electric motors are very quiet and have no need for transmission. The drawbacks of electric motors on the other hand are the poor battery lives and the high battery cost. Another drawback of the electric motors is that unlike their fossil-fueled brothers, electric motors require control circuitry, which increases the complexity of the drivetrain. Therefore this project was started to make it easier to design the drivetrain of electric vehicles or other products that make use of electric motors. This is to be accomplished by creating a circuit board for an educational platform in the form of a small autonomous car.

1.2 Purpose

The purpose of this project is to create a unified signal interface that can be used to power most types of electric motors. This report should have all the information required to design an electric drivetrain for any size of electric motor.

The report will answer the following questions:

- What components need to be replaced when switching up to a motor with a different voltage?
- What problems occur when designing ESCs?
- How does one solve or work around such problems?
- Which motors are most efficient?

1.3 Scope

This report covers the design of a circuit board controlling a 14.8V BLDC motor, design of its controlling software and guidelines of how to upgrade it to higher voltage motors. The report will only briefly explain how other types of motors work, not how to implement them. MOSFETs are covered in this report as the means to controlling the motors though there are other ways of doing it.

1.4 Specifications of the project

The circuit board designed in this project uses an STM32F103RB MCU and a DRV8303 MOSFET driver for a BLDC motor rated for 1.6A and 14.8V. The circuit board is outfitted with voltage regulators for the MCU and optional steering servo and tachometers. The circuit board can be used for controlling a steering servo, can be outfitted with tachometers for externally keeping track of the rotor position in the motor, but can also make use of internal hall effect sensors included in most motors. The circuit board supports CAN-communication for control signals and diagnostics, It also supports UART for debugging.

The controlling software will use a general signal interface to make it a “plug-n-play” module where the rest of the car does not need any setup to utilize it. This also assists in making the drivetrain scaleable. The MCU is programmed in C++.

2. THEORY

Electric motors make use of electromagnetic fields to generate rotating forces in their rotors. There are many ways to accomplish this, the most common types of electric motors are described below.

2.1 Brushed DC motors

One of the simplest electric motors. The Brushed DC motor consists of 4 parts: rotor, stator, commutator and brushes. The rotor is the rotating part which is surrounded in copper windings to turn it into an electromagnet, the stator surrounds the rotor and generates a constant magnetic field around it using magnets or electromagnetic windings. The commutator and brushes work together to allow the internal magnetic field to switch polarity when the motor is moving. The brushes are usually made of graphite and are prone to wear while sliding over the commutator.

The commutator and brushes constantly changing polarity of the windings in the rotor means there is no need for a control circuit when driving a brushed DC motor. All it needs is direct current flowing through it. However the brushes wear out eventually and are troublesome to switch out. Another problem with the commutator and brushes is that they generate sparks when switching which can be very dangerous in flammable atmospheres.

[2]

2.2 BLDC motors

Brushless DC motors, unlike their brushed brethren, make use of a changing magnetic field around the rotor instead of changing the polarity of the electromagnet in the rotor. This means that there is no need for commutators between the rotor and stator but also that they need permanent magnets which are expensive.

To drive a BLDC motor a controller circuit is required in order to switch the electromagnetic fields around the rotor in the right order and timing. This makes controlling a BLDC motor difficult, though also rewarding as they are generally more efficient than any other electric motor[1]. The control circuit driving a BLDC motor is known as an inverter and in the case of three-phase motors consists of 6 MOSFETs.

2.3 Induction AC motors

When describing induction AC motors there are two main types: synchronous and asynchronous AC motors. The difference between the two are that synchronous AC motors require the rotor to rotate at the same speed as the electromagnetic field while the asynchronous motors allow for “slip” between the electromagnetic field and the rotor.

Synchronous AC motors and BLDC are very similar in their construction except for one major detail: the stator windings in a synchronous AC motor are sinusoidal while the windings in a BLDC motor are trapezoidal. This means that the back-EMF produced by the motor will be sinusoidal for the synchronous motor and trapezoidal for the BLDC motor. The sinusoidal back-EMF will want a sinusoidal input while the trapezoidal type will want a trapezoidal input. A sinusoidal signal naturally comes from an AC source while a trapezoidal signal would be more suited to come from a DC source. [3]

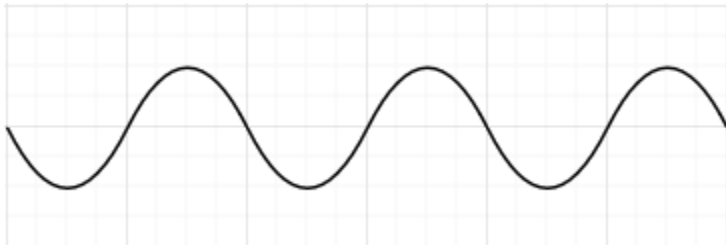


Figure 2.3.1: Sinusoidal signal

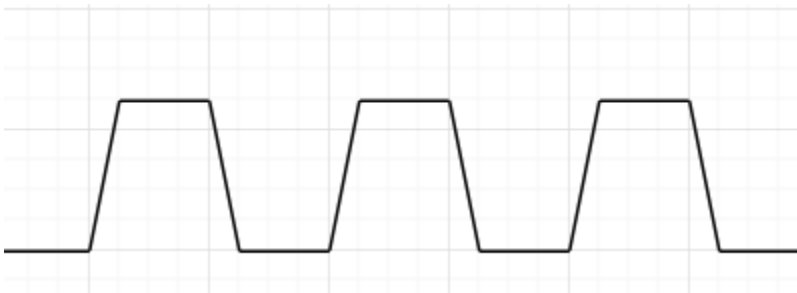


Figure 2.3.2: Trapezoidal signal

One very useful property of three-phase motors is that if two phases swap place, the direction of the motor is reversed. For a controller, that simply means switching place with two variables in the sequence.

2.4 Eagle

Eagle is a CAD-software by Autodesk which is used to design electrical schematics and PCBs. Components can be made in Eagle and added to a private library of components. When

components are made, a PCB footprint are also made to be able to place them out accurately when designing the PCB later. When designing the PCB, Eagle offers debugging to check if components or tracks are placed too close to each other. In conclusion, Eagle is very similar to both KiCAD and EasyEl which has been used in previous courses.

3. METHOD

This chapter covers the methods used to fulfill the specifications stated in chapter 1.

3.1 Procedure

The first step in designing the circuit will be to research how to control a BLDC-motor and then create a block diagram with all the necessary components and ports. At the same time in this stage what components to use will be decided, such as microcontroller, MOSFET driver and so on. The decision on what components to use is done parallel to designing the block diagram, so that the reference design can be kept in mind during the design.

When the block diagram is finished, it will be used as the basis to design the circuit schematic. With the help of the components reference designs and design choices, a circuit schematic that corresponded to the block-diagram will be designed.

The next step will be to design the PCB layout. The biggest issue to have in mind during PCB design is EMC. EMC could create noise in signal wires that could in a worst case scenario incapacitate the functionality of the PCB. Other than that is the issue of impedance in the wires if they are too long or thin, which could result in power dissipation and heat emission.

The next step after the PCB is designed will be to start with the design of the software. The already existing signal interface from the car will be used. Software for the MCU that will be designed will utilize this interface and act out all the commands it receives from other components. The computation will be executed inside the MCU to be able to control the units the drivetrain controls.

While waiting for the PCB to be delivered the most of the report will be written and the software coded.

3.1.1 Pilot studies

In order to get a grasp of what the scope of this project might include, research about electric motors and how to control them will be made.

Knowledge of what procedures to take when designing the circuit, how much resources will be need to finish the design (time, research materials and money) and how to manage the communication between the circuit and the rest of the vehicle. All things that will make working with this project more fluid.

The pilot studies are scheduled to take 4-5 weeks before initializing the electronics design and further studies will be made near the end of the project regarding what design properties need to be changed when using the circuit for different motors.

3.1.2 Electronics design

The circuit will be supplied with power from a 4-cell lithium battery at ca 14.8V to power the motor. For the logic circuitry a supply of 12V is transformed down to 3.3V. Input and output signals from the rest of the network include: requested motor speed, current motor speed, requested steering angle, current steering angle and several diagnostics signals.

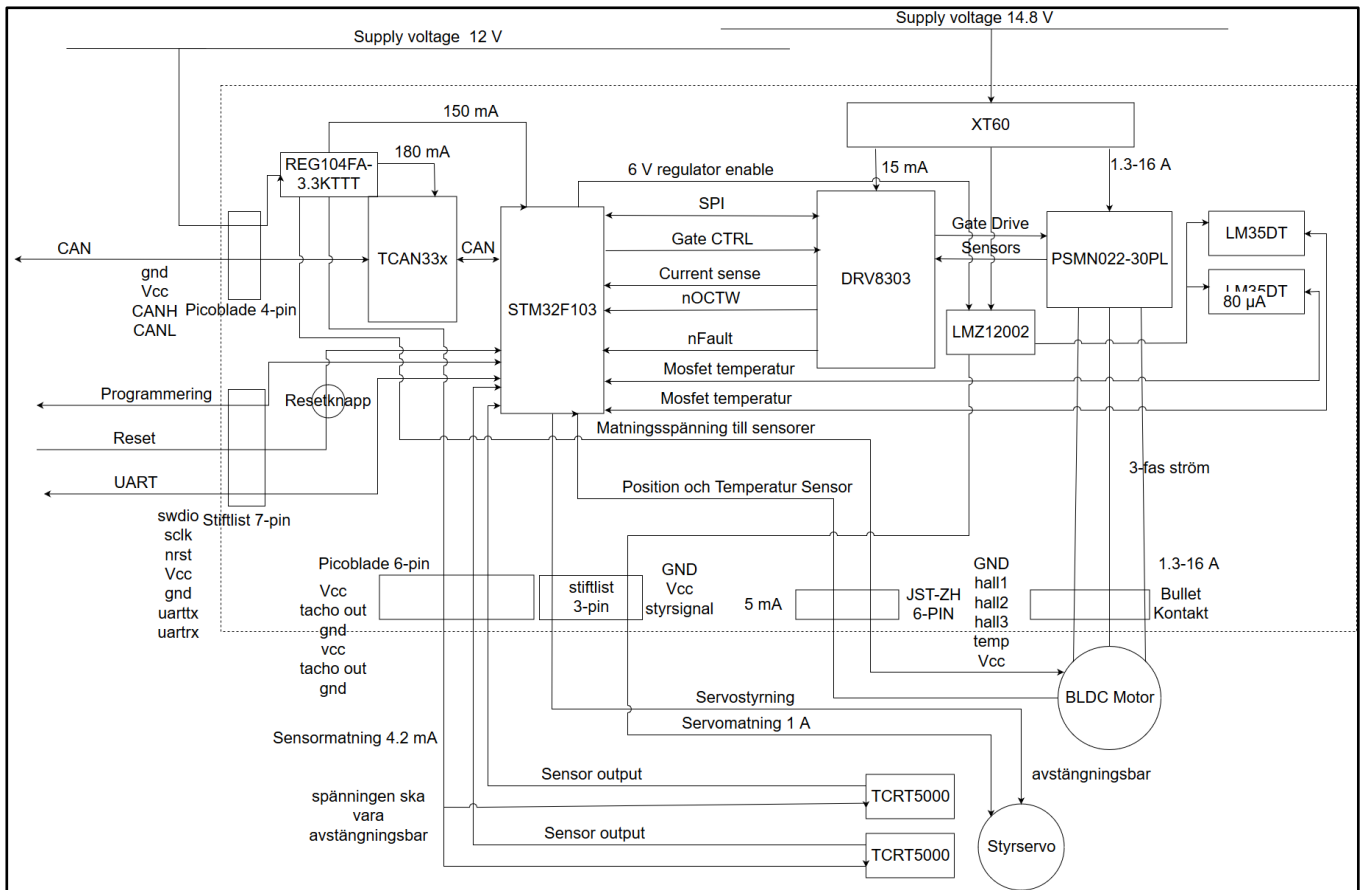


Figure 3.1: Electric driveline overview

Previous circuit designs at Infotiv have made use of STMf103CB and similar MCUs so one from the same series of MCUs will be used in this project as well. The main problem when designing the circuit is that the voltage used by the motor is much greater than the voltage used by the MCU and other ICs. To avoid problems with EMC and make changing components easier the low voltage part of the circuit is separated from the high voltage.

To drive the motor correctly, a three-phase output with alternating currents is required. To control these three phases, an inverter is needed. An inverter for a three-phase motor consists of six MOSFETs that can open and close in sequence to create alternating current. The problem with driving MOSFETs is that they require the gate signal to be of a voltage above the voltage that is at the source. Because the voltage used by the MCU is at 3.3V and the voltage supplied to the motor needs to be around 14.8V, a gate driver between the MCU and the MOSFETs will be needed. Because designing a MOSFET driver is not within the scope of this project, a premade IC will be used: DRV8303.

The network within the vehicle the circuit will be used in communicates using CAN (Controller Area Network), so an IC will be needed for that as well. Finally, IC's for voltage regulation must be chosen. One for transforming the 12V logic supply to 3.3V and one for transforming the 14.8V power supply to 6V for use with a steering servo and tachometers, both of which are optional and can be turned off.

When designing the schematic the reference designs of the ICs will be used and all design choices will be calculated. When choosing what non-IC components (e.g. resistors, diodes, transistors, etc) to use with the design, the most important factor is the cost and that they fit the specification.

Figure 3.2 shows the modules connected to the ESC. The ESC is in the bottom right corner and will be connected to the DC-DC module located at the top, second from the right. The DC-DC module regulates the voltage from the battery to 12V for the ESC and 5V for the other modules. The ESC is also connected through CAN to the central electronics module (CEM), located in the middle, second from the left. Other modules on the board are: telemetry module (TEM), battery management system (BMS) and advanced driver-assistance systems (ADAS).

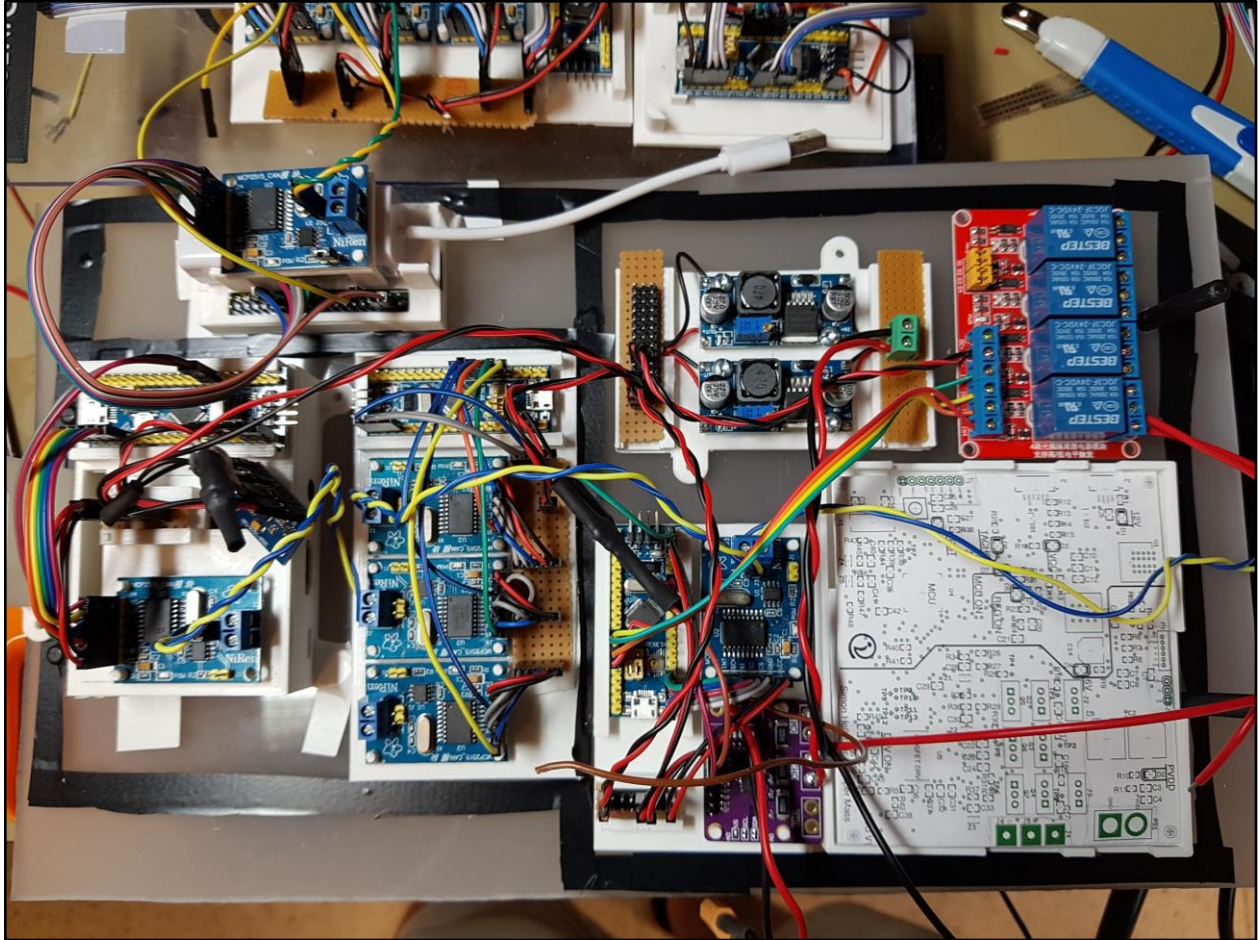


Figure 3.2: Inside of car

3.1.3 PCB design

All schematics for the PCB are made using Eagle. Design of the printed circuit board needs to be taken into account while working on the electronic schematics. Mainly choosing what pins to use for what signals in the MCU. Avoiding crossing wires is important to make the PCB design easier in the future.

When starting up the PCB design, all components will be arranged to the side of the circuit board with airwires connecting them. Airwires simply show what connections need to be made for each component. The most important part of the PCB design is to get a good layout. Useful guidelines are to leave a lot of room around the ICs with tight pins like the MCU and MOSFET driver and to have the ICs as spread out as possible to keep the temperature from rising too much in a concentrated area. Keeping the MOSFETs close to the power supply to minimize the electromagnetic fields generated by the high current flowing through them is also important.

Electromagnetic compatibility (EMC) is an important property to take into account when working with high currents. On this circuit, the main problem will be keeping the three phases isolated from the low voltage control signals as much as possible to avoid noise.

3.1.4 Programming of MCU

The MCU is programmed in C++ and the code is transferred to the chip using serial wire debug (SWD). Before the PCB is delivered and the design of the motor control software can fully start, preparations will be made by writing the software for smaller modules of the code that can be tested on a smaller PCB available containing an MCU of the same series.

The difference between C and C++ are that C++ is an object oriented language which gives access to classes. Classes in C++ are like structs in C but can hold functions. Classes are used to organise functions and the data the functions are meant to operate with. When creating classes you can set private or public access to members of the class. The members with public access can be accessed from outside the class. The members with private access on the other hand can only be accessed from within the class, this is the default access when creating members.

When the PCB is delivered, CAN communication and PWM will already be available and only the parts that require the PCB will remain (motor control, calibration, etc).

3.1.5 Testing

After the PCB is delivered, several tests will be made to make sure everything is working as intended. Primarily any obviously misplaced components will be investigated. Next, the MCU will be programmed. If no problems arise when programming it, all functions will be tested one at a time.

3.2. Research

A considerable amount of time was spent learning about the different types of motors and how they are controlled. One good resource that has plenty of information of all aspects of electric motors is *Electric Motors and Drives* by Austin Hughes[4]. More information that was very relevant to this project was found in *SMALL ELECTRIC MOTORS* by H. Moczala, J. Draeger, H. Krauß, H. Schock and S. Tillner[5].

To get an idea of how to design the circuit, the work of Benjamin Vedder[6] and his VESC (Open Source Electronic Speed Control) project was examined. Vedder's project has schematics

for the electronics and a finished PCB schematic that is free to use, but not much documentation. A similar MOSFET driver as the one in his project was chosen.

For tips on the important things to take into account while designing PCBs, a tutorial by David L. Jones was used[7]. The tutorial covers things such as layout recommendations for components, recommended minimum clearances between tracks and more.

Some time was also put into researching how bigger motors are controlled to make sure the drivetrain designed in this project will be scalable and applicable in future iterations.

4. DESIGN

The following chapter covers the design process of the PCB.

4.1 Designing the PCB

The design of the PCB began with an electric overview (see Figure 3.1) to get an idea of what signals were needed. Next, the long process of choosing appropriate components began. The MCU chosen needed to have an appropriate amount of pins to handle the amount of signals required and also the functions needed, e.g. PWM, CAN, and UART. Other ICs such as the voltage regulators, MOSFETs, gate-drivers, CAN module and temperature sensors had to be examined and reviewed to see if they would work with the design. An example is the MOSFETs chosen. They had to be rated for the same voltages and currents that would be used by the motor. Another important aspect of the MOSFETs are their ability to resist overheating. MOSFETs with metal lips protruding from them allow for mounting of heatsinks which help a lot. A gate-driver had to be chosen to accommodate the voltage threshold where the MOSFETs turned on. To make the design process faster, application examples provided by the manufacturers of all ICs were used. Other non-necessary components were also added e.g. LED lights to show what signals were active and also pads where users could watch signals with the use of an oscilloscope. After several design reviews with Infotiv, a schematic was finalised so work on the PCB design could begin (see Figure 4.1).

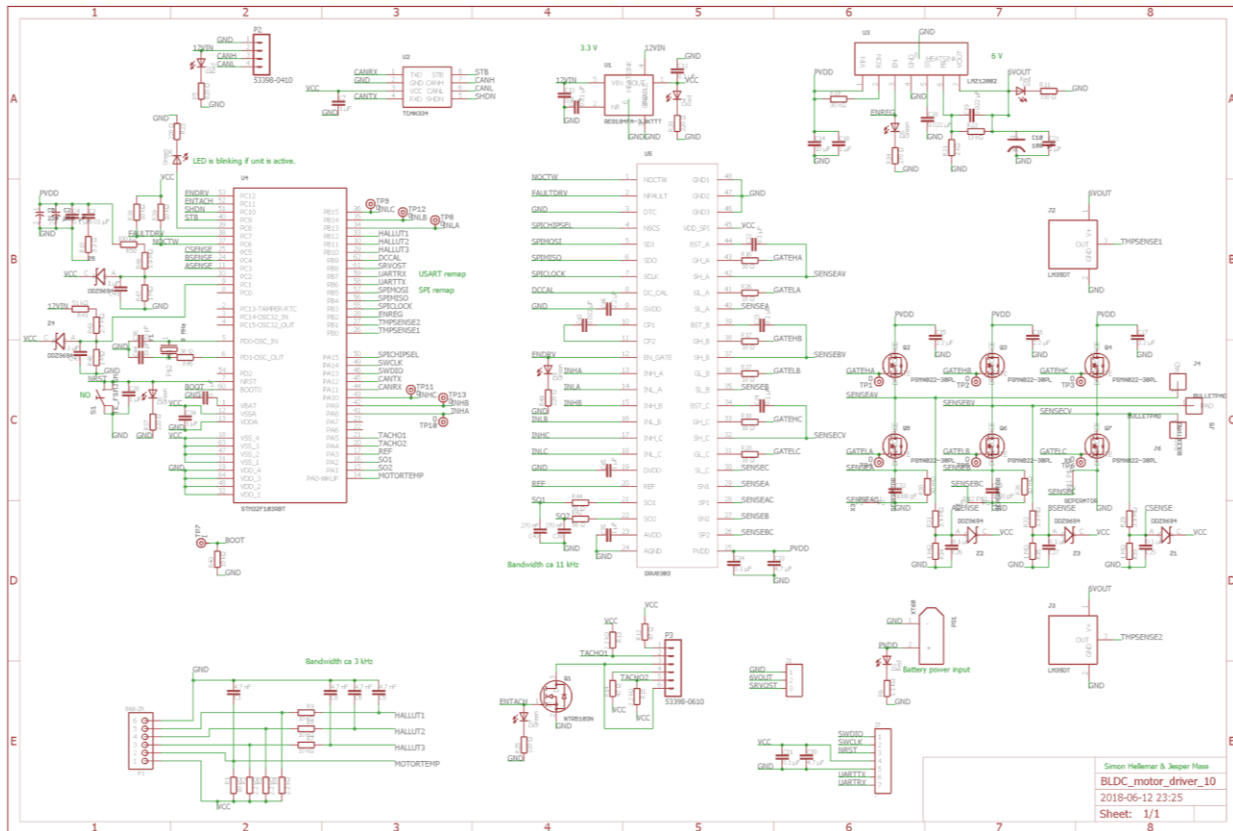


Figure 4.1: Electrical schematics

The limitations were a board size of 100x100mm, two layers to work with and only being able to put components on one side. This seemed unreasonable at first, but in the end it proved to be quite possible. The most important and lengthy part of the PCB design was to set the layout of the components. A full rework was needed at one point during development because all connections could not be made using the set layout. The design platform available at Infotiv is called Eagle.

One of the major objectives when designing the PCB was to keep the power tracks for the motor as far away from the logic wiring as possible to avoid noise in the signals from the high currents and the electromagnetic fields produced by them. The only tracks that are close to the power tracks are the control signals for the MOSFETs. After all components were connected, the remaining space on the board was filled with copper to create a large ground plane.

Figure 4.2 shows the full PCB design. Red areas are copper on the top of the circuit board while blue areas are copper on the bottom. Green details on the board are vias, holes in the circuit board that are laced with copper to allow current to pass to the bottom of the board. White details are on silk layer of the board, which means that they serve no function except to have text or other notes on the board.

In order to keep the MOSFETs cooled, heat sinks were screwed onto the lips of the MOSFET packages. The temperature sensors: J2 and J3 (see figure 4.2), would optimally be placed in-line with the MOSFETs: Q2 to Q7, but there was not enough room on the board for such a configuration. The temperature sensors will need extensions of their lips to the heatsinks of the MOSFETs in order to accurately measure their temperature.

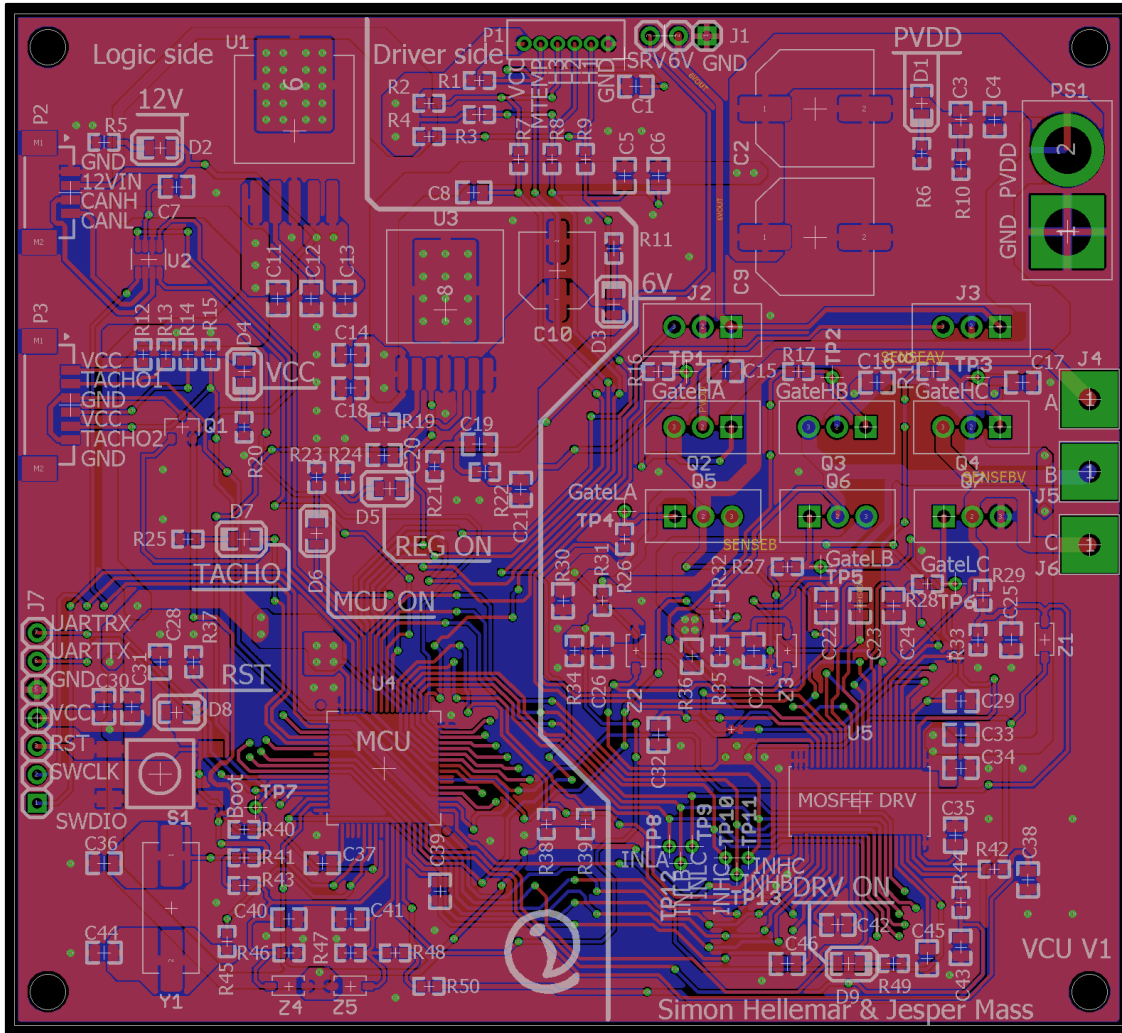


Figure 4.2: PCB design

The board is divided into two parts, the logic side and the driver side. As can be seen in figure 4.2, the left side is the logic side, with voltage regulators, MCU, CAN and more. To the right of the white line is the driver side with the MOSFETs, MOSFET driver and decouplers for the power input (C4, C2 and C9). Decouplers reduce noise in the power circuits. The diodes' (components named D#) functions are noted on the board, such as indication of when the MCU or driver is turned on. U1 and U3 are the voltage regulators the board. U1 regulates the 12V input down to 3.3V used by the MCU and CAN module. U3 regulates the power input which usually is around 14.8V down to 6V which is used by the tachometers and steering servo.

4.2 Solution alternatives

Because of the limited time this project had to produce the circuit, full optimisation was not possible. There was also not enough time to experiment with using different motor types with the inverter. Using relays for the inverter was examined, but relays are generally too slow to switch at the required frequencies, according to inspected datasheets and employees at Infotiv.

The choice of components for the PCB in this project could also have used some more work, such as calculating acceptable tolerances for the capacitors and resistors. Optimisation could also have brought down the total cost of the PCB as well as the heat generated from the components.

If a bigger PCB was used it would have been easier to separate the power parts of the board and the logic parts. It would also make sense to build two separate PCBs for logic and inverter. These solutions would have given the PCB a better electromagnetic compatibility (EMC). This is more important on higher voltages and bigger motors though.

5. RESULTS

The main purpose of this project is to produce a guide on building an electric drivetrain. The guide is divided into three areas: Motors and power, transistors and drivers as well as control.

5.1 Motors and power

The first area to discuss is the choice of motor and batteries or other power sources. Only BLDC motors are covered in this guide. When choosing motors one must first know how powerful it needs to be. What matters most is the power rating of the motor. The battery used in this project is a 4-cell LiPo (Lithium polymer) battery rating at 14.8V. The power supply for the logic circuitry should be separate from the supply to the motor for safety reasons. The logic circuitry should not be shut down because of a blown fuse or if the battery needs to be cut off.

5.2 Transistors and drivers

This project made use of MOSFETs to control the motor. MOSFETs are fast and can be used in a quite broad span of voltages but require the voltage of the control signal to be higher than the voltage at the source. This makes controlling MOSFETs troublesome with low voltage logic circuits without some sort of driver between the MOSFETs and MCU. Therefore, using a premade IC is recommended. MOSFETs are only capable of working within certain voltages and can only transmit as much current as they are rated for before overheating, so those are the main things to take into account when choosing MOSFETs.

When choosing driver, the description of the drivers usually informs of what voltage levels it was designed for. Most MOSFET drivers have different means of control so reading the datasheet thoroughly is required to properly set them up for use in a circuit. A lot of drivers have built-in security measures such as shunt current protection and overheating protection which improves the overall safety of the circuit board.

When working with motors for full scale electric cars, individual drivers for each transistor is often needed but they're still controlled the same way.

5.3 Control

To simulate a three-phase signal, controlling six MOSFETs is necessary. The MOSFETs need to be opened in sequence to allow current to flow into the motor from one of the three at the top of figure 5.3.1 and out of the motor from one of the three at the bottom in order to get the current to

flow into one winding and out of another. This generates electromagnetic fields that will be able to drive the motor.

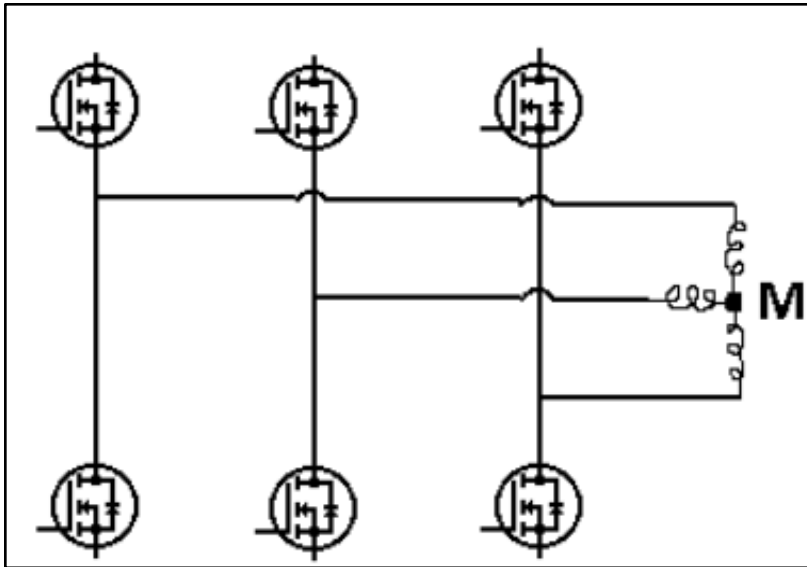


Figure 5.3.1: Inverter

In order to get the sequence right, some type of position sensing is required from the motor. Most small BLDC motors include integrated hall effect sensors which can sense the motor position quite accurately. Other ways of sensing rotor position include measuring back-EMF from the windings and tachometers. In this project, the motor included hall effect sensors which were used. Though the circuit also supports use of tachometers.

Sensing the rotor position and knowing what signals to send is not all. To control the MOSFETs, knowledge of how the MOSFET driver uses the incoming signals is also necessary. Controlling the MOSFET driver is different for most drivers, but the most common way is using PWM signals. The reason for using PWM signals is because most MCUs are able to produce them while simultaneously running other bits of code. Changing the duty cycle of the PWM signal will change the current flowing through the motor and thus also the torque. In table 5.3.1 the result of the signals sent to the MOSFET driver are shown. Note how sending an “on” signal to both the high and low MOSFET of a phase sets both the gates to low. This is to prevent short circuiting the phase. The driver used in this project also has built in dead-time to prevent shoot through currents.

Table 5.3.1: 6-PWM mode driver inputs and outputs

INH_X	INL_X	GH_X	GL_X
0	0	L	L
1	0	H	L
0	1	L	H
1	1	L	L

To get the motor running at requested speed, a PID controller function will be used. The controller will regulate the switching frequency of the inverter to get the motor up to speed. In order to reverse the motor, the function for switching phases in the inverter will change the direction of the rotor by switching two phases in the sequence.

5.4 Setup in varying scales

The current design is made for a 11V motor, 16V battery and a 12V logic power buss.

When installing the drivetrain some things needs to be changed depending on the power rating of the motor. If there is a 12V power buss, there is no need to change any of the components on the logic side of the printed circuit board (PCB), except those connected to the battery. These are the resistors in the voltage divider that measures battery power. If using a bigger power source than 16V the 6V regulator for the thermometer and steering servo will be needed to be changed for a more powerful one. This also applies when changing the servo to a more powerful one which needs more power. Unless more pins are needed there is no need to change the microcontroller unit (MCU). If the motor does not use hall-effect-sensors the components for that will not be needed, and those pins on the MCU will be free and can be used for other purposes.

On the Driver side nothing needs to be changed if using an 11V motor and a 16V battery. If a more powerful motor is used, the resistors for the LEDs will have to be changed to accommodate the higher voltage. If using a motor below 60V, the gate driver will suffice. If a greater one is used, the gate-driver will have to be changed to one rated for that power. The same goes for the metal oxide semiconductor field effect transistors (MOSFETs) which will suffice if below 30V and 22A is used for the motor. If a much greater motor is used, using insulated-gate bipolar transistor (IGBT) in its place should be considered. When working with higher voltage most passive components may be need to be changed to components with higher power rating. When choosing a new driver, the use of pins on the MCU will probably vary. There are MCU:s with more pins from the same series, these are suggested to be used because they are programmed the same way and have the same functionality except with more pins. With greater motors the MOSFET/IGBT sometimes use individual drivers.

All variables that needs to be changed in the code will be found in a Calibration file.

6. CONCLUSIONS/ DISCUSSION

Electric motors come in many shapes and controlling all of them with a single circuit is plausible, but would require a lot of components. Especially considering the large span of voltages available. The circuit board constructed in this project is made specifically for controlling a three-phase BLDC motor, but could easily be reprogrammed to control a brushed DC motor as well since it only requires a single output. Controlling an AC induction motor with the circuit would be possible but it would be much less efficient and weaker than using a controller made specifically for such motors. Mainly because AC induction motors have sinusoidal windings which are more compatible with sinusoidal signals. To make a sinusoidal signal, a filter would be required.

When working with bigger motors, the MOSFETs will need to be switched out for MOSFETs with higher voltage ratings. The same goes for the MOSFET driver, though drivers can usually function over broader voltage ranges than the MOSFETs, so they do not need to be switched out as often. Most premade MOSFET driver IC's for high voltage are designed to be connected to one MOSFET each, unlike the driver used in this project which connects to all six MOSFETs. The voltage regulators will also need to be exchanged to accommodate for the higher voltages. It is also recommended to keep the MOSFETs further away from the logic part of the circuit or even keep them on a separate circuit to avoid problems with electromagnetic fields generated by high currents used by bigger motors.

Another problem when working with bigger motors is the increased heat emission from the MOSFETs as higher currents flow through them. Using MOSFET packages with lips or similar protrusions allow mounting of heat sinks which usually is enough to keep the MOSFETs cool. In some cases, active cooling such as air cooling or water cooling might be necessary to keep the circuit board from overheating. Using MOSFETs with high efficiency can alleviate this problem slightly. Another way of keeping the general heat down on the board is to keep the bigger components spread out to avoid concentrated heat generation.

In conclusion, designing an electric drivetrain has many variables that need to be accounted for. To make designing the circuit board a feasible task, a thorough specification of requirements is needed. The most important factors are what voltage the motor requires, how much current the motor will be using at maximum load and what signal interfaces needs to be taken into account. After those two factors are figured out, some design parameters need to be set up, such as: cost, space, precision and temperature. After this, the design can begin.

REFERENCES

- [1] S. Bistak and S. Y. Kim, “AC Induction Motors vs. Permanent Magnet Synchronous Motors”, 2017. [Online]. Available: <http://empoweringpumps.com/ac-induction-motors-versus-permanent-magnet-synchronous-motors-fuji/>, Accessed on: 11th May 2018
- [2] R. Condit, “Brushed DC Motor Fundamentals”, 2010. [Online]. Available: <http://ww1.microchip.com/downloads/en/AppNotes/00905B.pdf>, Accessed on: 11th May 2018
- [3] D. Collins, “What’s the difference between BLDC and synchronous AC motors?”, 2015. [Online]. Available: <https://www.motioncontroltips.com/faq-whats-the-difference-between-bldc-and-synchronous-ac-motors/>, Accessed on: 17th May 2018
- [4] A. Hughes, *Electric Motors and Drives: Fundamentals, Types and Applications*, 3rd ed., Oxford, United Kingdom: Newnes, 2006
- [5] H. Moczala, J. Draeger, H. Krauß, H. Schock and S. Tillner, *SMALL ELECTRIC MOTORS*, English edition, London, United Kingdom: The Institution of Electrical Engineers, 1998
- [6] B. Vedder, “VESC - Open Source ESC”, 2016. [Online]. Available: <http://vedder.se/2015/01/vesc-open-source-esc/>, Accessed on: 18th May 2018
- [7] D. L. Jones “PCB Design Tutorial“, 2004. [Online]. Available: <http://alternatezone.com/electronics/files/PCBDesignTutorialRevA.pdf>, Accessed on: 3rd June 2018