TECHNICAL UNIVERSITY OF DENMARK
INSTITUT FOR ELEKTROTEKNOLOGI

31015 - INTRODUCTORY PROJECT
ELECTROTECHNOLOGY

ROAST TELEMETRY

Authors:	*Student ID*
HØRDUR ANDREASEN	s173933
SIGNE NORBROG	s174030
BJARKE GRINDER VALHØJ	s173406

Sunday 21st June, 2020

Abstract

We were tasked with designing a telemetry system that would communicate between two moving vehicles for DTU Roadrunners solar-car project. Due to the COVID-19 lock down in Denmark, several of the design stages were simplified, so the final solution was a compromise between what we would have liked to implement and what would be feasible. Two solutions are considered in this paper; a custom STM32 PCB design, and a Teensy development board prototype. We discuss advantages held within each design, and determine the next course of action for the Telemetry project.

12.1

1 Introduction

The world solar challenge is an international solar vehicle race spanning 3000 kilometres through the center of Australia. A subsection of DTU Roadrunners, the roadrunners solar team abbreviated as DTU ROAST, has begun production of a cruiser class solar car for the 2021 race. The race is a feat of engineering, as the most energy efficient electric vehicles must cross Australia on public roads with a tenth of the energy a regular car requires to cross the same distance. The remainder of the energy is entirely supplied by the sun.
A significant issue with driving through the center of Australia, one of the hottest places on Earth, in a solar powered vehicle, is that both batteries and solar cells are very flammable and can combust if not properly maintained. As safety is a necessity for participants and bystanders, there is a strong desire to be able to remotely monitor the solar vehicle in a supporting vehicle, which may be up to 500 metres away. This is to ensure that an additional failsafe is added to the vehicle. In the event the driving team is occupied and automatic measures malfunction, the supporting vehicle can act as an extra set of eyes throughout the race.
An additional benefit of remotely monitoring the vehicle, is that it enables innovative data analysis solutions. A ROAST member mentioned applying model predictive control in the supporting vehicle, optimizing the solar vehicle speed to ensure maximum energy efficiency. The issue with this method is the hardware and computational power required would significantly diminish the return if placed in the solar vehicle. Herein lies the Telemetry project, which would enable two-way communication between the solar vehicle and the supporting vehicle, allowing data transfer up to a theoretical distance of a 1000 metres line of sight. The premise is that relevant data from the Controller Area Network bus (CAN-bus) is sent to the supporting vehicle, and the support vehicle crew have the option of returning commands to the solar vehicle.
The DTU ROAST project is still in its infancy, which has posed significant challenges for us as several key design considerations related to the telemetry module have yet to be determined. This has lead to a very flexible design philosophy where we are forced to reconsider large portions of the design to adapt to new requirements. At the time of writing, the module is still in ongoing development, and the current paper is a review of the design process thus far.

2 Problem statement

Remotely measuring the condition of the solar car during operation provides valuable insight into safety and performance. As the driver is often occupied with the vehicle itself, observers in supporting vehicles can ensure important components are operating within parameters, or provide recommended course of action for the driving team. There are several technical and managerial challenges given the current stage of the DTU ROAST Project, as such we will investigate the following statements:

- How can we construct a system which allows for automatic and accurate transmission and receiving of sensor data, through a wireless medium, on moving vehicles up to 400 metres apart?
- How can we construct a system that is modular enough so that it can be applied by the solar-car team, without them needing to apply considerable changes to the system?
- Can we optimise the system so that the transmission does not significantly impair the performance of the solar-car?

3 Planning

The project spanned 13 weeks from the 3rd of February to the 11th of May, and three weeks from the 8th of June to the 22nd of June. The project was subdivided into four sprints of four weeks each, with a meeting with DTU ROAST at the end of each sprint. The intention was that each sprint would be a significant step to the final product, with the meetings providing major feedback guiding the next sprints' course of action. The sprints were planned as follows;

- Preliminary design proposition
- Design stage 1 (Develop chosen design further)
- Design stage 2 (Polish design, CAD work)
- Production, testing & paper

The intention was that through iterative design stages and feedback from DTU ROAST, it would be possible to construct a functional prototype in the final sprint. Due to external circumstances, this would not come to fruition. At the start of our third sprint, COVID-19 forced the country into lockdown which significantly limited the ability to progress at the desired pace. Most notably was the delay in communication, as we shifted from in-person to email discussions, what could be discussed in a few hours took several days as replying was a slow process. We decided not to advance to the production stage, as uncertainties in the design and lack of facilities would result in a sub-optimal product.

4 Design sprints

In this section we briefly cover the purpose of each design sprint and the major design decisions taken throughout.

Preliminary design

The initial design sprint could be described as the wild west of ideas. The project was not set in stone at this stage, and project specification varied wildly as we discussed what functions our module should provide. We made the following high-level designs for the sprint meeting:

- A complicated design which could provide advanced functionality to the central processor of the car, allowing both long and short range communication interfaces for support vehicle and nearby phones. Ref. fig 1
- A simple stand-alone module which only provided long-range communication to support vehicle and a tyre pressure monitor receiver. Ref. fig 2

It was decided that our module should be simple and reliable, as critical processes could be reliant on it. Therefore we opted for a simpler stand-alone module, and were asked if it would be possible to handle wireless tyre pressure monitoring and storage of sensor data in a black box.

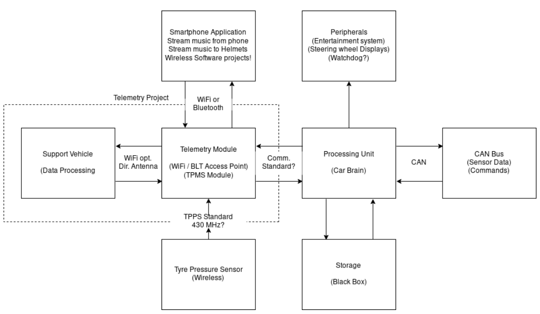

Figure 1: Complicated design

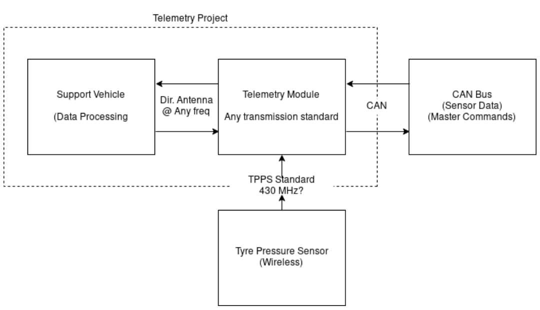

Figure 2: Simple design

Design stage 1

With an initial project specification, we began development of an electrical diagram and an initial choice of components. It was decided early on that we would like to construct a single PCB with integrated IC and RF transceiver modules, as it would be a valuable learning opportunity and a first for all of us. We opted to use an STM32F0 IC as the center of our design as we had prior experience with it, and it came with an integrated CAN controller, USB, and low current consumption.

A Teensy 4.0 development board was suggested as an alternative to the STM32. It has long been in use with the Roadrunners ecocar, and several other ROAST members were using it, allowing for mutual coding practices. We will in the later stages compare our STM32 design with the Teensy, as both designs have their drawbacks.

We opted not to include a black-box setup into our module as data is already sent from the car and would further complicate the design. There was discussion with the software team that the central processor would display data to nearby phones via Bluetooth, something which would require data storage as is. We concluded that it would be best if this data-storage was used as a black-box. In the event that either module malfunctions, there would still be sensor data gathered for later analysis.

Design stage 2

An hour after the prior sprint meeting, COVID-19 lockdown was in effect. The adjustment to working from home was difficult, and work pace slowed significantly. We made the following adjustments to our design to reduce design complexity and mitigate the time lost from COVID-19;

- RF-Module: Originally we wanted to build our own integrated RF-transceiver, but were advised to use an RF-Module (nRF24+LNA+PA) as a cheap solution to get a functional prototype. This reduced design complexity significantly.
- Tyre Pressure Monitoring System (TPMS): We inquired for weeks regarding the wheel type, as it is instrumental for choosing an appropriate TPMS receiver. After a significant time span, we were told that no wheels had been chosen and this feature was impossible to implement at the current stage, as it requires reverse engineering a receiver module, which varies greatly between wheel types. We opted to delay this feature until appropriate wheels had been chosen.

Given these two simplifications to the design, we spent the remainder of the sprint going back and forth between Electrical diagrams and PCB CAD drawings. We discovered an oversight in our design with the STM32 integrated CAN 2.0 module. Being unfamiliar with CAN standards, it was only late into the design stage that we discovered we should support the latest CAN standard (CAN-FD). We opted to implement a CAN-FD Controller IC using the remaining SPI port of the STM32, which inflated our Bill of Materials more than we originally anticipated.

Final sprint

We originally hoped to get the design into production at this stage, but review has proven to be an important and slow process. Adjustments to the PCB were made to ensure reliable performance, mitigating high frequency noise and ensuring each component is decoupled properly. As we lack the facilities to construct the STM32 design during COVID-19 lockdown, we have explored the possibilities of a Teensy 4.0 design.

5 RF-Module

We opted to use an nRF24 module with a built in power amplifier for transmission, and a low noise amplifier for reception. This is a standard cheap RF-module advertised to support reliable transmission up to a distance of a 1000 metres line of sight. This has been reliably proven by the open source community [1], although it requires the following modifications to the module:

- Decoupling and Voltage regulation: The module is rated at 3.3 V and has a high current draw during transmission. We ensure a reliable voltage regulation and plenty of decoupling capacitors to ensure proper function.
- Shielding: Wrapping the module in an insulating layer, and aluminium foil connected to the ground pin, reduces the noise floor, extending reach.

Applying these steps, we successfully established a communication link of approximately 500 metres in a rudimentary test. Additional testing is required as both amplifiers were configured incorrectly, but currently we believe this is a good foundation for further development. An important question asked during sprint meetings was the effect of the Doppler shift; Does the velocity of the vehicles affect the signal link? The conclusion is no, the frequency shift experienced is due to relative motion between transmitter and receiver. Assuming both cars maintain approximately the same velocity, frequency shift is minimal. [2]

$$f_{rx} = f_{tx} \frac{v_0 + v_{rx}}{v_0 + v_{tx}} \quad (1)$$

6 KiCad design

A significant portion of the lockdown was spent doing CAD work on our STM32 design, as little else could be done at the time. Diagrams can be found on our Github repository here. We will briefly cover specific parts of the CAD process which we found interesting.

Component choices

The benefit to ST Electronic ICs is the thorough documentation specifying every component in their datasheets and application notes in great detail. Little work was required to search for components, and to reduce BoM size the same components are used repeatedly across the entire PCB.
To further reduce costs, we chose components we had available at home or through DTU. A downside to this is breaking design conventions, an example is a large footprint diode, which takes up as much space as the ICs.

High frequency design

Special care is required for high speed digital signals. Since our signals are low frequency, design conventions are merely a recommendation, not a requirement. An example would be USB 2.0 differential traces, which state a 90 Ohm impedance matching. We opted to minimize the trace length, as impedance matching was not available for a two-layer board. [3]
The crystal oscillator used, requires a separated ground plane, as shown in Figure 4. This follows the design convention stated in application note AN2867 [4]. Notice ground vias are placed near components, often using several vias to reduce inductance and parasitic capacitance, in turn reducing the high impedance paths to ground at high frequency. [5]

Power

We discovered that the USB voltage supply line is very noisy, additional filtering was required to mitigate high frequency noise from impacting the performance of the circuit. We added a ferrite bead in series with the linear voltage regulators decoupling capacitors to make a low pass filter.
The powering scheme used in the designed board is a very rudimentary "source multipoint" [5], a daisy chain formation used by novices as it is very easy to implement. It is the least power efficient distribution scheme, and we were recommended to attempt a "star" formation. This requires a total redesign of our PCB, something we will consider in the next iteration.

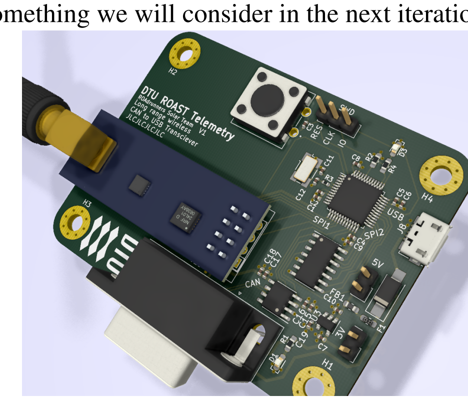

Figure 3: KiCad Render of current PCB design

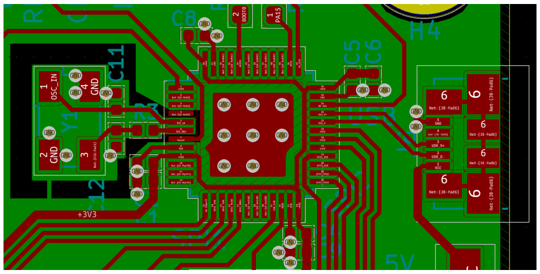

Figure 4: Cutout of PCB design: Notice the USB differential trace, separate ground plane for oscillator and amount of ground vias

7 Coding

Programming the project consisted of creating a transmitter code and a receiver code. The transmitter code had to be able to read data from sensors, such as temperature or pressure sensors, and then send this data to the receiver. The receiver code had to be able to read the data sent from the transmitter, as well as process it. The code shown in Algorithm 1 is a very simplified Pseudo code for the Transmitter. A pseudo code is a simplified version that shows the transmitter sending data, as well as checking for a connection to the receiving module. When there is a connection to the receiver, it will print an auto acknowledgement to help with debugging.

```
Result: Sending Data
initialization;
while Loop do
    Listen for data;
    if Received data then
        Stop listening for data;
        Store data;
        Send data;
        if No connection from the receiving NRF24
          then
            Print No Acknowledgement;
        else
            Print Acknowledgement;
        end
    else
        Keep listening;
    end
end
```

Algorithm 1: Pseudo code for the Transmitter

The code shown in Algorithm 2 is also a simplified version of a pseudo code for the receiving module. It shows, that if the receiver is able to make a connection to the transmitter, then the receiver will read the data, store it, and then display it.

```
Result: Receiving data
initialization;
while loop do
    if radio is available then
        Read data;
        Store the data;
        Print the data;
    end
end
```

Algorithm 2: Pseudo code for the Receiver

Both the STM32 and the Teensy have good options for programming. The STM32 has its own software program that can be used, which makes it easy to write a program for the specific use of the STM32 microchip as well as pairing up the pins. The Teensy, although it does not have its own software, can be used and programmed with the Arduino integrated development environment (IDE). The overall programming of this project, once created, will be applicable to either the Teensy or the STM32, since they are both programmed using the C programming language, though with the need for small adjustments such as pin connections and libraries used.

8 Discussion

With the application of a modified nRF24 module, we are guaranteed line of sight transmission up to 1000 metres. This exceeds the original specification of 400 metres and gives additional leniency in the event of bad weather or excessive signal fading. The current issue is what should drive the nRF24 module, an STM32 custom PCB or a Teensy 4.0 development board? In this section we will discuss our current designs and their drawbacks.

Modularity

The benefit of a development board is the simplicity of the design and the relative ease to make modifications if required. The Teensy is more than powerful enough for our usage, and additional features can easily be added, given the pins required are available. As the Teensy has seen stable use in Roadrunners projects, there is a vast collective knowledge base to draw from when it comes to programming.

The STM32 design is entirely custom made, and as such is inherently inflexible when produced. Additional features require a new design and manufacturing costs. The added complexity with a custom PCB makes it difficult to get into, but once understood adds far more flexibility compared to the Teensy.

Current consumption

Despite the complexity, the STM32 design has the potential to be very low power, and can be further reduced through hardware and software optimization. The Teensy has a current draw of 100 miliamperes at maximum capacity, which can be significantly reduced in software by clock scaling and sleep timers. We cannot conclude which would be more power efficient for the solar car unless measurements are made on both designs, but our belief is that the custom made PCB can be designed to be lower than the Teensy.

Costs

The important question here is how many modules we construct. The Teensy development board has a fixed price of approximately 250 DKK per module, excluding any PCB breakout boards. The price of a STM32 board inversely scales with production, the shipping cost of PCB being a significant portion of the price. At 10 modules constructed, the price would be 150 DKK per module.

Verdict

The competition is based on energy efficiency, as such we believe a power optimized custom PCB is desired in the long run. We believe the STM32 design is a good foundation for further development. Additional time should be spent revisiting the choice of microcontroller and features we opted to simplify or exclude. In the short term we believe a Teensy based proof of concept would provide valuable data for further comparison, and should have been a priority early on the development stages. We opted to proceed with CAD work early as it was the only way to work during COVID-19 lockdown according to the initial plan, but in hindsight a breadboard prototype should have been a part of the initial plan to begin with.

9 Conclusion

We conclude that we managed to design a system which should support a stable communication link beyond the specified 400 metres distance. Due to improper planning and external circumstances additional time is required to construct a prototype and revisit features we chose to omit. We believe the long-term goal should be a highly power efficient custom PCB, moreso than the requirement for modularity, a requirement we decided on due to the uncertainty experienced early into the project development. The road to this goal is a longer process with several iterations, this design being a stepping stone, as it is currently beyond our level as electrical engineers.

Acknowledgement

We as a group are grateful for the help received from DTU ROAST. The design meetings provided us with extensive professional and educational guidance, and with the help of their challenging questions, we were able fix some overseen errors. We would especially like to thank Christian Kampp Kruuse for his quick responses to our emails. With his help we were able to gain insight in things that we did not have a lot of experience with. We would also like to thank Vitaliy Zhurbenko for our brief discussion in regards to the wireless portion of the system. We would like to thank Tiberiu Gabriel Zsurzsan as well for his help in reviewing our PCB design and providing numerous insights into PCB design in general. Lastly we would like to thank our supervisor Claus Suldrup Nielsen who has provided us an excellent learning experience with this project, and has ensured we were thoroughly welcomed in DTU Roadrunners. With all our hearts we thank you for all the beer.

References

[1] Oitzu (24/02/2016) *Fixing your cheap nrf24l01+ pa/lna module* [online] Found at: `<https://hackaday.com/2016/05/31/fixing-the-terrible-range-of-your-cheap-nrf24l01-palna-module/>` [online] Available at: `<https://web.archive.org/web/20180106003715/http://blog.blackoise.de/2016/02/fixing-your-cheap-nrf24l01-palna-module/>`

[2] LibreTexts Physics (March 2020) *17.8 The Doppler Effect* [online] Available at: `<https://phys.libretexts.org/Bookshelves/University_Physics/Book%3A_University_Physics_(OpenStax)/Map%3A_University_Physics_I_-_Mechanics%2C_Sound%2C_Oscillations%2C_and_Waves_(OpenStax)/17%3A_Sound/17.08%3A_The_Doppler_Effect>`

[3] Silicon Labs (January 2013) *AN0046: USB Hardware Design Guidelines*. [pdf] Available at: `<https://www.silabs.com/documents/public/application-notes/an0046-efm32-usb-hardware-design-guidelines.pdf>`

[4] ST Electronics (January 2020) *AN2867: Oscillator design guide for STM8AF/AL/S, STM32 MCUs and MPUs*. [online] Available at: `<https://www.st.com/content/st_com/en.html>`

[5] Texas Instruments (November, 1999). *PCB Design Guidelines for Reduced EMI* [pdf] Available at: `<https://www.ti.com/lit/an/szza009/szza009.pdf>` [Accessed June 2020]

DTU•Elektro	Spring 2020	Group: 12 ID: MEK-2
Course 31015	Title: SOLAR CAR Electronics	Group members
Introductory Project - Elektroteknology	Telemetry for Solarpowered Car	Bjarke Valhøj, s173406 Hørdur Kai Andreasen, s173933 Signe Norborg, s174030
Document:	Project Statement	1
Version/Status:	Final	Draft

Problem statement

Measuring the condition of the solar car while it is operating is valuable in not only determining its performance, but also ensure the safety of the drivers. Since the driver might be too occupied while the vehicles is running it is important that the measurements can be observed by someone else. Due to this it is important to investigate the following points listed by priority:

- How can we construct a system which allows for automatic and accurate transmission and receiving of sensor data, through a wireless medium, on moving vehicles up to 400 metres apart?
- How can we construct a system that is modular enough so that it can be applied by the solar-car team, without them needing to apply considerable changes to the system?
- Can we optimise the system so that the transmission does not significantly impair the performance of the solar-car?