Bachelor's thesis

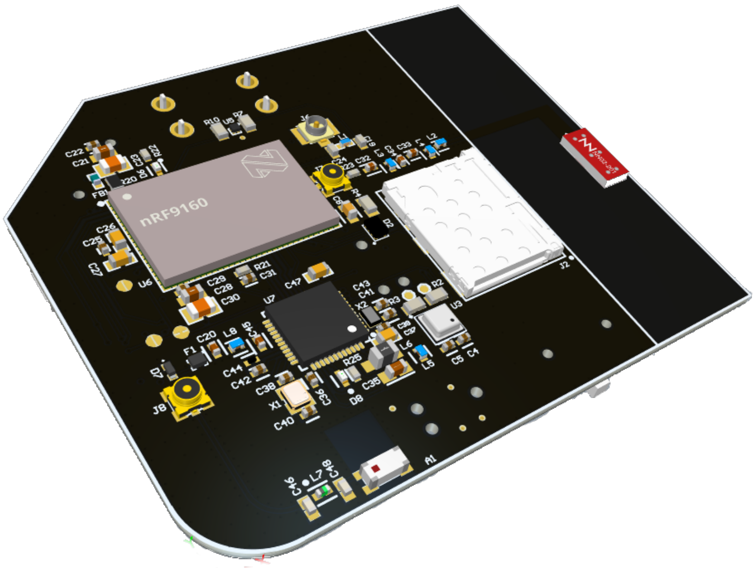
NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Electronic Systems

Marius Hammer Olaisen
Lars Fredrik Sveine Onsager
Njål Rundereim
Jonathan Haugen Tønnesen

Cellular IoT Asset Tracker

IELET2900 Bachelor's thesis in Electronics and Sensor Systems

Bachelor's thesis in Electronics and Sensor Systems
Supervisor: Rolf Kristian Snilsberg
May 2023



NTNU
Norwegian University of Science and Technology

NORDIC
SEMICONDUCTOR

Marius Hammer Olaisen
Lars Fredrik Sveine Onsager
Njål Rundereim
Jonathan Haugen Tønnesen

Cellular IoT Asset Tracker

IELET2900 Bachelor's thesis in Electronics and Sensor Systems

Bachelor’s thesis in Electronics and Sensor Systems
Supervisor: Rolf Kristian Snilsberg
May 2023

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Electronic Systems

NTNU

Department of Electrical engineering

IELET2900 - Bachelor Thesis Electrical Engineering

Cellular IOT asset tracker

Authors:
Marius Hammer Olaisen
Lars Fredrik Sveine Onsager
Njål Rundereim
Jonathan Haugen Tønnesen

Supervisor:
Rolf Kristian Snilsberg

May, 2023

NTNU

Tittelside Bacheloroppgave BIELEKTRO

Oppgavetittel (norsk og engelsk):

Cellular IoT Asset Tracker
Mobilbasert IoT-eiendelssporer

Forfattere: Olaisen, Marius Hammer Onsager, Lars Fredrik Rundereim, Njål Tønnesen, Jonathan Haugen	**Prosjektnummer:**	E2316
	Innleveringsdato:	22.05.2023
	Gradering:	[x] åpen [ ] lukket

Studium:	Elektroingeniør
Studieretning:	Elektronikk- og sensorsystemer
Veileder internt:	Rolf Kristian Snilsberg
Institutt:	Institutt for elektroniske systemer
Oppdragsgiver:	Nordic Semiconductor
Kontaktperson:	Joakim Tønnesen

Sammendrag (norsk og engelsk):

Ved å utvikle på Nordic Semiconductor sin nRF9160-SiP, skulle vi lage et IoT-system med fokus på batterisparing. Ved å anvende strømsparende teknikker innenfor både hardware og software, ble det utviklet et system for detektering av MOB-hendelser på sjøen. Rapporten tar for seg detaljerte valg mellom ulike komponenter for å bygge to forskjellige kretskort som skal utgjøre systemet, som kan kommunisere både internt med Bluetooth og eksternt via LTE-nettverk. Teori om antenner, transmisjonlinjer, og GNSS blir introdusert. Resultatet er et proof-of-concept som må videreutvikles på de faktiske prototypene.

En spesiell takk rettes til Joakim Tønnesen og Kostiantyn Denysenko ved Nordic Semiconductor for all hjelp.

By utilizing Nordic Semiconductor's nRF9160-SiP, the group aimed to create an IoT system focused on battery conservation. By using power-saving techniques within both hardware and software, a system for detecting MOB events at sea was developed. The report addresses detailed choices between various components to build two different circuit boards that will make up the system, capable of communicating both internally via Bluetooth and externally via the LTE network. Theory about antennas, transmission lines, and GNSS is introduced. The result is a proof-of-concept that needs further development on the actual prototypes.

Special thanks go to Joakim Tønnesen and Kostiantyn Denysenko at Nordic Semiconductor for all their assistance.

Stikkord norsk: nRF9160, nRF52832, Thingy:91, Nordic Semiconductor, LTE-M, BLE, CoAP, PCB, GNSS, Zephyr RTOS, MOB-deteksjon, prototype	**Stikkord engelsk:** nRF9160, nRF52832, Thingy:91, Nordic Semiconductor, LTE-M, BLE, CoAP, PCB, GNSS, Zephyr RTOS, MOB Detection, prototype

Table of Contents

List of Figures

List of Tables

Glossary

AOI Automated Optical Inspection. A non-contact testing method used to check the quality of printed circuit boards. It uses a high-speed camera to capture an image of the board and compares it to the correct (golden) image to detect any defects.. 44

BeiDou A Chinese satellite navigation system providing global coverage, intended for both civilian and military applications. 27

BGA Ball Grid Array. A type of surface-mount packaging used for integrated circuits. 23

BR/EDR Basic Rate/Enhanced Data Rate - a mode in Bluetooth primarily used for continuous wireless connections, for example in audio streaming. Also known as Bluetooth Classic.. 41

callback In programming, callbacks are functions that are passed into another function. They are 'called back' or executed at a later time, typically in response to certain events. 42

EMI Electromagnetic Interference, a disturbance generated by an external source that affects an electrical circuit through either electromagnetic induction, electrostatic coupling, or conduction. 13

Galileo A global satellite navigation system developed by the European Union, designed to provide global coverage and improved accuracy. 27

Generic Attribute Profile (GATT) An architecture that specifies how data is transferred between Bluetooth Low Energy (BLE) devices. GATT uses a concept of services and characteristics. A service is a collection of related pieces of user data, defined in a hierarchical manner, and each service contains one or more characteristics. Each characteristic can hold different types of data, such as user data, configuration data, etc. These characteristics are identified by UUIDs. 42

GLONASS Global Navigation Satellite System operated by Russia, providing global coverage for both civilian and military purposes. 27

GNSS Global Navigation Satellite System, a system that uses satellites to provide autonomous geospatial positioning. 27

GPS Global Positioning System, a satellite-based navigation system operated by the United States. 27

Kconfig A configuration system used in the Linux kernel and other projects to define and manage build-time configuration options.. 3

LNA Low Noise Amplifier, an electronic amplifier used to amplify weak signals while keeping the noise level as low as possible. 27

PCA Printed Circuit Assembly, a process in which electronic components are assembled and soldered onto a printed circuit board, forming a functional electronic device. 43, 44

PCB Printed Circuit Board, a board that mechanically supports and electrically connects electronic components using conductive tracks, pads, and other features etched from copper sheets laminated onto a non-conductive substrate. vi, 6, 7, 11, 25, 43, 56, 59

PIFA Planar Inverted-F Antenna, a type of antenna with a compact design, commonly used in mobile devices due to its low profile and multi-band capabilities. 11, 27, 53

QFN Quad Flat No-Lead. A type of flat, square or rectangular surface-mount plastic package with leads located on its perimeter. The "no-lead" descriptor refers to the fact that the leads are not extended, but instead wrap around the edges of the package. 25

SDK A set of software tools and libraries that developers can use to build applications for specific platforms or devices.. 3

semaphore A programming construct used to control access to a common resource by multiple processes in a concurrent system such as a multitasking operating system. It acts like a signal or a flag, coordinating actions between different tasks or threads to avoid conflicts. 32

STL Standing for stereolithography, is a file format used primarily in 3D printing and computer-aided design (CAD). It represents three-dimensional objects by defining the surfaces using a mesh of small triangles. These triangles are defined by the coordinates of their vertices and an outward-facing normal. This format is "geometry-only", meaning it doesn't include any information about color, texture, or other attributes common in more complex CAD models. STL files are widely used for their simplicity and ease of use in slicing software for 3D printing, as well as in numerical simulations for engineering analyses. However, the simplicity also comes with limitations, such as the inability to represent curved surfaces precisely.. 43

UUID Universally Unique Identifier. In the context of Bluetooth, a UUID is a value used to uniquely identify specific types of services provided by a device. Bluetooth devices assign a unique 128-bit UUID to each service they offer, enabling seamless communication and interaction between compatible devices. 40

Chapter 1

Introduction

Group E2316 has conducted this bachelor thesis on behalf of Nordic Semiconductor with the goal of incorporating the nRF9160 chip into an asset-tracking device and optimizing it for extended battery life. The problem statement is provided in Appendix A. Cellular IoT asset trackers have gained significant popularity as monitoring tools for valuable assets [1]. These trackers are often small, low-power devices that utilize cellular networks to send information such as temperature, humidity, location, and other sensor readings, to a cloud-based platform for real-time analysis and visualization. Cellular IoT asset trackers have a wide range of potential applications, including logistics and supply chain management, fleet tracking, and equipment monitoring.

1.1 Problem definition

This thesis will concentrate on the challenges associated with using the nRF9160 chip for asset tracking in marine environments, where conditions may be less than ideal for reliable data transmission. The focus will be on addressing the issues encountered when developing a system for detecting Man Overboard (MOB) situations on boats, utilizing the nRF9160 chip both as a standalone component and in combination with other Nordic Semiconductor devices that interact locally.

The proposed system aims to overcome these challenges by taking advantage of the capabilities provided by the nRF9160 chip and other components within the Nordic Semiconductor ecosystem. The system will be designed to enable wireless communication both internally among the devices and with the cloud, all while optimizing battery life for improved performance.

1.2 Project goals

The overall project goal of this thesis is divided into three different sections. Each of these subsections defines different parts of the project and their goals.

1.2.1 Effect goals

These are the goals that benefit the company the most.

- **Proof of concept** is the main goal for Nordic's interest in this project. Nordic is planning to add more products to what they already offer so they can show them to potential customers. Having more systems utilizing their chips will help with that goal.
- **Validation of the systems** Since Nordic is a continuously growing company, they are always looking for ways to validate and document their already existing products.

- **New implementations** Along with validating their already existing products, Nordic is looking for new ways to implement their products.
- **Expansion to new markets**. As with any expanding company, Nordic is always looking for new ways to sell their products to new customers. By having the group produce a new system, they can hopefully show customers that the concept works and they can increase their sales.

1.2.2 Result goals

These are the goals that the group, along with Nordic, has defined as the finished product.

- **Low power sensor module** is the main end goal for this product. The group has decided to call the unit the *main unit*, which is the unit that is carried on person while out at sea.
- **Communication over Bluetooth and LTE** is the main communication standard the system will utilize. The Nordic nRF lines of products are made specifically for these types of communication.
- **Small and portable**. The device should be small and portable for ease of use.
- **Waterproofing** is another key feature of the system. Since the system is intended to be used at sea, waterproofing will be key. This will mostly come through in the casing design, but sensors and components should accommodate a waterproof unit.

1.3 Nordic Semiconductor Product Portfolio

Nordic Semiconductor offers a diverse range of products designed for low-power Internet of Things (IoT) applications. Some of the key products in their portfolio include their SiP (System in Package) nRF9160, and their SoCs (System-on-Chips) nRF52840, and nRF52832.

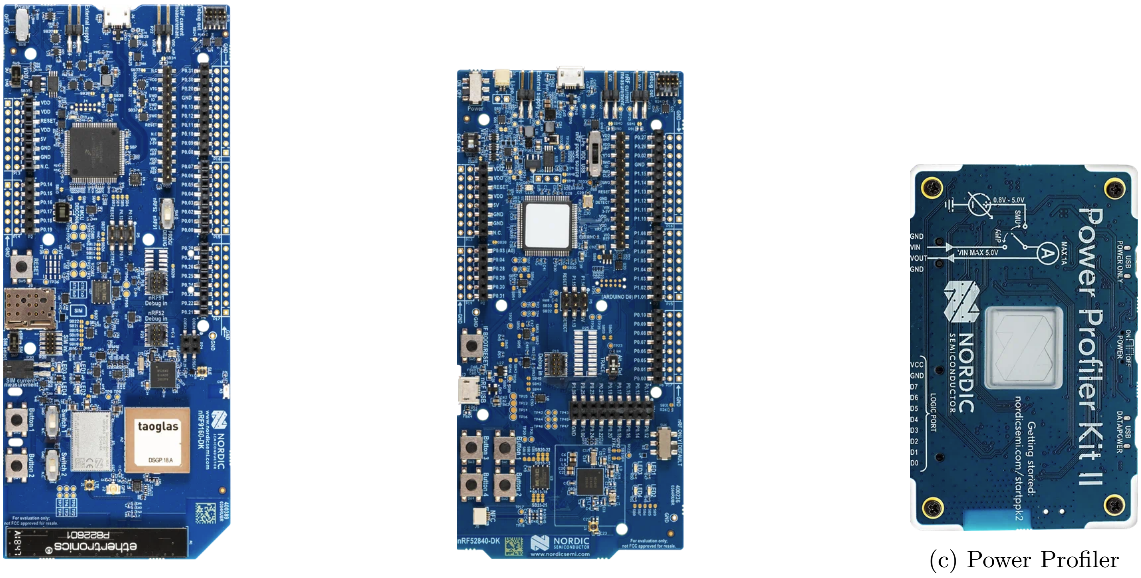
The nRF9160 is a highly integrated SoC specifically designed for cellular IoT applications, featuring a multi-mode LTE-M/NB-IoT modem, GPS, and various other features. It is versatile, easy to integrate, and ideal for battery-powered and secure applications. Appendix B shows an overview of the features of this chip.

The nRF52840 SoC is a powerful, highly flexible device with support for Bluetooth Low Energy (Bluetooth LE), ANT, Zigbee, Thread, and other wireless protocols. The nRF52832 SoC is another versatile solution designed for Bluetooth Low Energy and other wireless applications, offering great power efficiency and performance.

To aid in the development of IoT solutions, Nordic Semiconductor provides several development kits and tools, such as the nRF9160 development kit [2], nRF52840 development kit [3], the Thingy:91 prototyping platform [4], and the Power Profiler Kit 2 [5]. These development kits offer platforms for exploring the full capabilities of the respective SoCs, simplifying the development process for IoT applications. The Thingy:91 is an easy-to-use development platform with a pre-certified cellular IoT module based on the nRF9160 SoC, ideal for rapid prototyping and proof-of-concept projects. The Power Profiler Kit 2 is a versatile tool for measuring and optimizing power consumption in IoT devices. Some of these products are shown in Figure 1.1.

1.3.1 Nordic Developer Academy

The Nordic Developer Academy is an online platform that offers training courses and learning resources for developers working with Nordic Semiconductor's technologies [6]. These courses cover what is needed to start working on the development kits, including wireless communication protocols, software development tools, and best practices for designing and implementing solutions based on Nordic's products. The group members will complete three courses provided as of writing this thesis, which are:



(a) nRF9160 DK

(b) nRF52840 DK

(c) Power Profiler Kit II

Figure 1.1: Some of the products developed by Nordic Semiconductor

nRF Connect SDK Fundamentals: This course focuses on the essentials of firmware development using the nRF Connect Software Development Kit (SDK). It covers topics such as the Zephyr RTOS, devicetree, Kconfig, and interfacing with common hardware peripherals and external sensors [7].

Bluetooth Low Energy Fundamentals: This course teaches the basics of Bluetooth LE using Nordic Semiconductor devices (nRF52 and nRF53 Series). The course explains how to create a Bluetooth LE prototype and establish wireless data channels between two Bluetooth LE devices [8].

Cellular IoT Fundamentals: This course covers the essentials of cellular IoT application development using the nRF Connect SDK. It includes topics such as LTE-M and NB-IoT technologies, software architecture of cellular IoT applications, and hands-on exercises for connecting to the cloud and using the GNSS receiver inside the nRF9160 SiP [9].

1.4 Our Solution

Our team aims to develop a comprehensive Man Overboard (MOB) system that utilizes Nordic Semiconductors Thingy:91 as a *vessel unit* and a wearable Main Unit with the nRF9160 for cloud communication and nRF52838 for local communication. A third unit with only the nRF52832 will also be produced. This system is designed to effectively address the issues outlined earlier by incorporating three key components, as shown in Figure 1.2.

- **Main Unit (nRF9160 & nRF52832)**: A wearable device capable of detecting when a person falls into the water and autonomously sending a distress signal. Will be able to function as a stand-alone unit. It is equipped with the nRF9160 chip for LTE communication and the nRF5238 chip for local communication with the Vessel Unit and other units.
- **Vessel Unit (Thingy:91)**: A relay unit placed on board the vessel, designed to receive distress signals from the Main Unit and transmit alerts to nearby vessels and authorities, by providing better signal in areas with limited coverage. This unit also includes a kill switch feature, which will stop the vessel's engine when prompted to.
- **Auxiliary Units (nRF52832)**: Additional devices that communicate with the Vessel Unit through Bluetooth. These units are designed to be able to function both on people as and on critical equipment on board for surveillance of critical equipment.

The primary objective of this solution is to demonstrate the potential for an advanced MOB system that enhances the overall safety of maritime activities, such as recreational boating, commercial fishing, and sailing, while maintaining power efficiency. By utilizing the capabilities of the Nordic Semiconductor chips, we aim to showcase a reliable and efficient solution that offers real-time monitoring, rapid response, and precise location data.

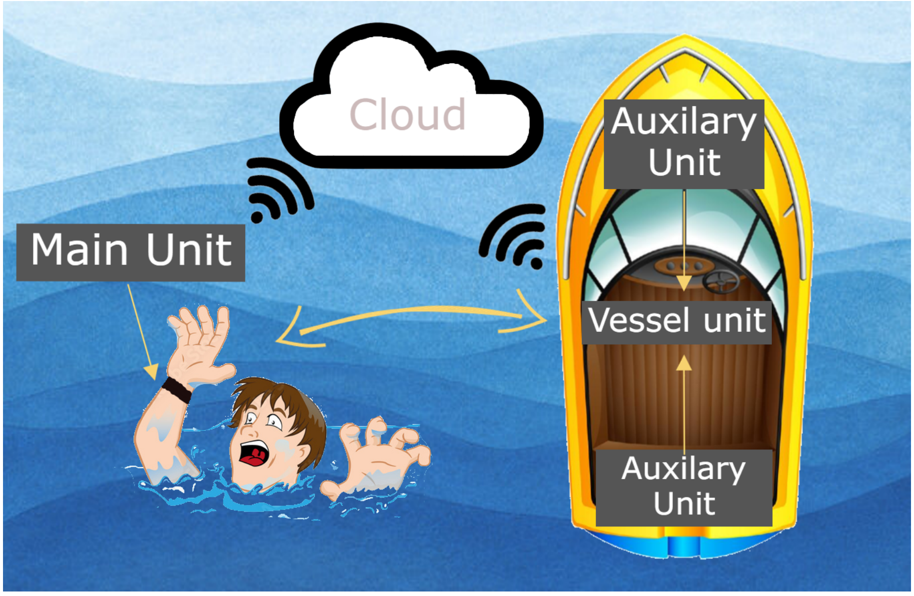


Figure 1.2: MOB system Implementation

As a proof-of-concept, the system's design will be based on modular architecture and scalability, allowing for the development of a minimum viable product that can be refined and improved over time.

Through the implementation of this system, we aim to demonstrate the potential of using the nRF9160 chip in a system for significantly reducing risks associated with man overboard incidents, as well as surveillance of equipment, increasing the likelihood of successful rescue operations, and promoting safer maritime practices.

1.5 The landscape today

In January 2023, Norway's Minister of Fisheries and Ocean Policy, Bjørnar Skjæran, called for a vision zero for drowning fatalities at sea, [10] which was supported by the Norwegian Seafarers Union's advocacy for mandatory use of personal AIS transponders [11]. AIS (Automatic Identification System) technology provides vessel tracking information and can quickly locate a person in the water in the event of an MOB incident by transmitting their location to nearby vessels and rescue organisations. If a law mandating the use of on-person AIS transponders is enacted, the Norwegian market for such products is expected to surge, with potential for international adoption.

1.6 Potential problems

Throughout the course of this bachelor thesis project, the team will need to address various challenges related to power saving, unit volume, debugging. In this section, we outline the main problems that the group will prepare to face and discuss possible strategies to overcome them. By proactively identifying and addressing these potential issues, we aim to ensure the successful completion of the project.

1.6.1 Power saving

Power saving is one of the main scopes of this thesis. nRF9160 and nRF5340 are low power hardware optimal for low power cellular asset trackers. The hardware is capable of different power-saving techniques and optimizations that have been implemented to enhance the performance and efficiency of the units. This section will give a brief introduction on power saving employed in the development process.

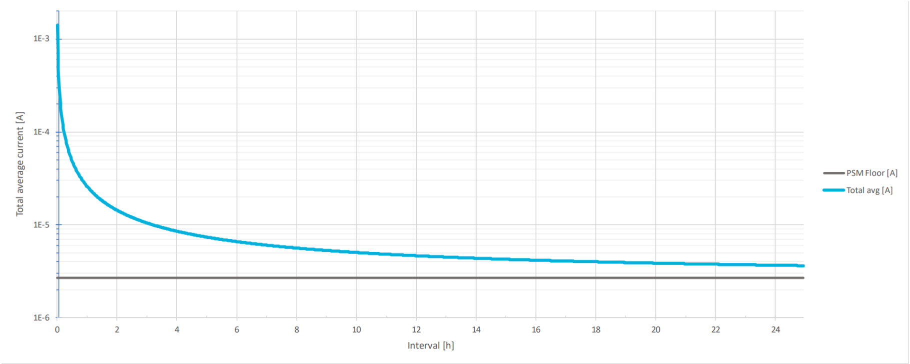


Figure 1.3: Battery efficiency on the nRF9160 [12]

Figure 1.3 illustrates the inverse relationship between power consumption and the interval duration for the unit to be active for data transmission and reception [12]. In terms of battery efficiency, it's crucial to maintain a low number of these intervals without compromising the required functionality. Since the majority of our data transmission will be uplink (UL), the need for the unit to frequently receive data may be minimal. Considering that the Vessel Unit is not a self-contained entity and will be linked to the boat's battery, power-saving measures will primarily be relevant for the wearable devices.

1.6.2 Unit volume

In relation to the unit volume it is important to carefully consider the circuit board design and pin-outs during the design process. By optimising the layout of the board and taking advantage of the available pin-outs for an external debugger, it is possible to create a more compact and streamlined asset tracker. This can result in a product that is more comfortable for the user and easier to use.

1.6.3 Debugging

Since the system comprises of two separate system-in-packages, they will need to be debugged separately. One potential challenge could be establishing a seamless connection between the two IC's across the SoCs. To tackle this, we will be utilizing two development kits from Nordic for debugging, which will allow us to debug both chips simultaneously and iterate through the code.

Chapter 2

Theory

The creation of a MOB system draws upon a multitude of engineering disciplines, encompassing Electronics, Power Electronics, Radio communications, Internet of Things (IoT), and Software Engineering. The theoretical concepts are presented in a concise manner and provide a broad overview. For a more in-depth exploration of the concepts, readers are encouraged to consult the referenced sources for further details.

2.1 PCB Design principles

2.1.1 PCB stackup

A PCB stackup consists of multiple layers of conductive material, typically copper, separated by insulating layers, or substrate, typically made of a material like FR-4 [13].

The four conductive layers usually serve specific purposes [14]:

1. **Top Layer:** Also referred to as the component or signal layer. It's where the majority of components are mounted and surface traces are routed. This layer is often involved in high-frequency signal transmission.
2. **Inner Layer 1:** Typically used as the power plane. This layer is usually dedicated to distributing DC voltages to the components mounted on the PCB.
3. **Inner Layer 2:** Usually serves as the ground plane. This layer provides a common ground for all electrical circuits on the PCB, and it's critical for signal integrity and power return.
4. **Bottom Layer:** Also a signal layer, it's used for component placement and trace routing, especially for through-hole components. It's also a place where additional traces can be routed if the top layer becomes crowded.

Figure 2.1 shows a visualization of a 4 layer PCB stackup. PrePreg [15] and FR-4 are types of composite materials , usually glass fiber, used as isolation between layers.

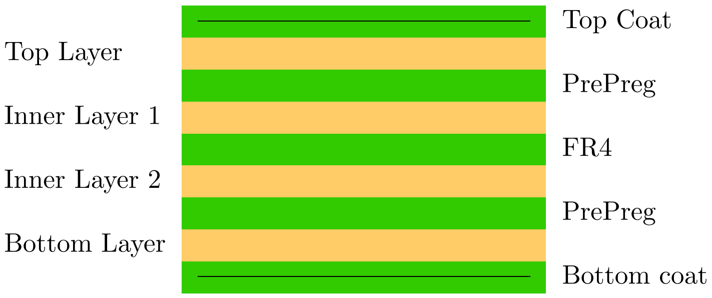


Figure 2.1: Visualization of PCB stackup

Each of the different copper layers has an internal resistance, dependent on different factors. This resistance is calculated using Equation 2.1. ρ is the inherent resistivity of the material, l is the length of the copper trace, and A is the cross-section of the trace

$$R = \frac{\rho \times l}{A} \tag{2.1}$$

Traces also act as a series inductance that can be calculated using Equation 2.2.

$$L = \frac{\mu_0 \mu_r A}{l} \tag{2.2}$$

Where:

- L is the total Inductance of the trace
- μ_0 is the permeability in a vacuum
- μ_r is the relative permeability
- A is the cross-sectional area of the trace
- l is the total length of the trace

The multiple conductive layers of the stackup, also make the PCB act as a capacitor. To calculate the generated capacitance, Equation 2.3 is used.

$$C = \epsilon_0 \epsilon_r \frac{A}{d} \tag{2.3}$$

Where:

- C is the overall capacitance between the traces
- ϵ_0 is the permeability in air
- ϵ_r is the relative permeability
- A is the cross-sectional area of the trace
- d is the distance between the layers

2.1.2 Transmission Lines

Transmission lines, sometimes referred to as Data lines[16], are used to transport electrical signals from one place to another. These structures are defined by parameters such as characteristic impedance (Z_0), attenuation constant, and phase constant.

The characteristic impedance of a transmission line is given by the formula:

$$Z_0 = \sqrt{\frac{R + j\omega L}{G + j\omega C}} \tag{2.4}$$

where R is the resistance per unit length, L is the inductance per unit length, G is the conductance per unit length, C is the capacitance per unit length, and $j\omega$ is the complex angular frequency[17].

2.1.3 Voltage standing wave ratio

Voltage Standing Wave Ratio (VSWR) is used to measure the efficiency of power transfer between a transmission line and an antenna. VSWR is calculated as the ratio of the maximum voltage to the minimum voltage of a standing wave on the transmission line, as shown in Equation 2.5. A low VSWR value (typically less than 1.5) indicates that most of the power transmitted along the line is being radiated by the antenna, while a high VSWR value (typically greater than 2) indicates that some of the power is being reflected back towards the transmitter. High VSWR can result in power loss, increased noise, and reduced system performance [18].

$$VSWR = \frac{1 + |s_{11}|}{1 - |s_{11}|} \tag{2.5}$$

2.1.4 Smith charts

The Smith Chart serves as a valuable tool for the analysis of transmission lines and RF components, offering a graphical depiction of complex impedance. The real component of impedance is represented along the horizontal axis, with the right extremity symbolizing an open circuit ($r = 1$) and the left, a short circuit ($r = -1$). Circles of constant resistance depict the imaginary part of the impedance, with the upper and lower halves respectively denoting inductive and capacitive impedances. The characteristic impedance of the transmission line, typically 50Ω, is signified by the chart's center [19]. The lower section of the chart illustrates reflection coefficients and VSWR.
When tuning the impedance of a transmission line, the impedance will move on the Smith chart. Depending on how and what components are placed along the transmission line:

- Capacitors will move the impedance down.
- Inductors will move the impedance up.
- Series Components will follow the path of constant real impedance.
- Shunt Components will follow the path of constant real admittance.

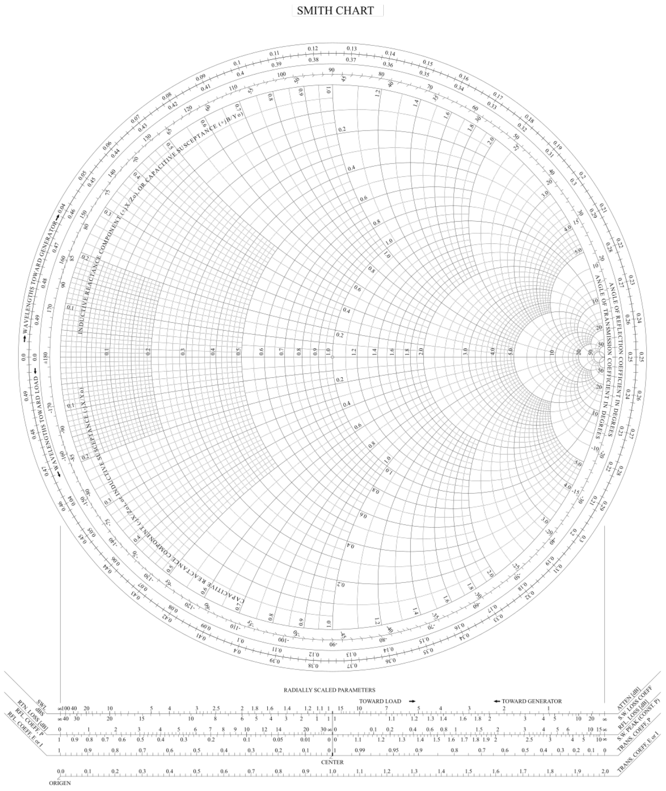


Figure 2.2: Basic Smith chart [19]

2.1.5 Parasitic properties

Parasitic properties, such as capacitance, inductance, and resistance, are unwanted effects that occur in circuits, impacting their performance. These properties can cause signal degradation, noise, power loss, and other issues in high-speed digital circuits, analog circuits, and power electronics.

To mitigate the effects of parasitic properties, designers carefully select materials, components, and circuit layout. They also use simulation and analysis tools for optimization, and techniques like decoupling capacitors, shielding, and ground planes to minimize the impact of parasitic properties [20].

2.2 Antenna

Antennas work by converting electric currents into electromagnetic waves (transmission) or electromagnetic waves into electric currents (reception). The conversion process is governed by Maxwell's equations, which describe the relationship between electric and magnetic fields and their sources.

2.2.1 Maxwell's Equation

Maxwell's equations are a set of four partial differential equations that describe the behavior of electric and magnetic fields [21]. They are given by:

$$\nabla \cdot \mathbf{E} = \frac{\rho}{\varepsilon_0} \tag{2.6}$$

$$\nabla \cdot \mathbf{B} = 0 \tag{2.7}$$

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t} \tag{2.8}$$

$$\nabla \times \mathbf{B} = \mu_0 \mathbf{J} + \mu_0 \varepsilon_0 \frac{\partial \mathbf{E}}{\partial t} \tag{2.9}$$

Where:

- $\mathbf{E}$ is the electric field
- $\mathbf{B}$ is the magnetic field
- ρ is the charge density
- $\mathbf{J}$ is the current density
- ε_0 is the vacuum permittivity
- μ_0 is the vacuum permeability

2.2.2 Electromagnetic Waves

Electromagnetic waves are the carriers of information in wireless communication systems. They consist of oscillating electric and magnetic fields that are perpendicular to each other and propagate in the direction perpendicular to both fields [22]. The speed of light (c) in a vacuum is the maximum speed at which electromagnetic waves can travel.

2.2.3 Radiation Pattern

The radiation pattern illustrates the spatial distribution of radiated electromagnetic waves. It provides valuable information about an antenna's performance and can be used to determine the optimal orientation for transmission or reception. The radiation pattern is typically represented using polar or Cartesian coordinate systems and can be analyzed in three dimensions or as two-dimensional cross-sections in the horizontal and vertical planes. [23]

2.2.4 Antenna Gain

Antenna gain quantifies an antenna's ability to focus radiated energy in a specific direction. It provides a measure of an antenna's directivity and efficiency. The gain is expressed in decibels (dB) relative to a reference antenna, typically an isotropic radiator or a dipole antenna. [24]

Isotropic Radiator

An isotropic radiator is a hypothetical antenna that radiates energy uniformly in all directions. It serves as a reference point for comparing the performance of real antennas. Antenna gain relative to an isotropic radiator is denoted as dBi [25].

Dipole Antenna

A dipole antenna is a common, real-world reference antenna for gain measurements. It consists of two conductive elements of equal length, separated by an insulator, and has a total length of approximately half the wavelength of the operating frequency. Antenna gain relative to a dipole antenna is denoted as dBd [26].

Calculating Gain

The gain of an antenna can be calculated using the following formula:

$$G = D \times \eta \tag{2.10}$$

Where G is the gain, D is the directivity, and η is the efficiency of the antenna. By incorporating both directivity and efficiency, the gain provides a comprehensive assessment of an antenna's performance. Higher gain values indicate better overall performance, enabling more efficient transmission and reception of electromagnetic waves. [27]

2.2.5 Impedance

Antenna impedance is a critical parameter that characterizes the electrical behavior of an antenna at its terminals. It is a complex quantity, consisting of a real part (resistance) and an imaginary part (reactance), and plays a vital role in the efficient transfer of power between the antenna and the transmission line or transmitter. [28]

Resistance

The real part of the antenna impedance, resistance, represents the power losses in the antenna structure due to the flow of current. Resistance can be divided into two categories: radiation resistance and ohmic resistance.

- **Radiation Resistance**: Radiation resistance is associated with the power radiated by the antenna as electromagnetic waves. It is a fictitious resistance, as it does not involve power dissipation in the form of heat, but rather accounts for the energy radiated into space.
- **Ohmic Resistance**: Ohmic resistance is due to the resistive properties of the antenna's conductive materials. Power losses caused by ohmic resistance are dissipated as heat within the antenna structure.

Reactance

The imaginary part of the antenna impedance, reactance, represents the energy stored in the electric and magnetic fields of the antenna. Reactance can be either capacitive or inductive, depending on the antenna's electrical properties and operating frequency.

- **Capacitive Reactance**: Capacitive reactance occurs when the electric field between conductive elements dominates the energy storage. It is typically observed in antennas operating below their resonant frequency.
- **Inductive Reactance**: Inductive reactance occurs when the magnetic field around conductive elements dominates the energy storage. It is typically observed in antennas operating above their resonant frequency.

2.2.6 Inverted-F antenna

An inverted F antenna, or PIFA is a type of antenna used in wireless communications. Figure 2.3 depicts an example of a PIFA. The length and height in the figure are matched to be $\lambda/4$ at the resonant frequency, which is 2.4GHz [29]for this antenna. The arrow marks the feed port, which should normally be adjusted to have an impedance of 50 Ω. The impedance is dependent on the PCB stackup [28].

The performance of the antenna can be affected by any copper or conducting material present underneath or on top of the antenna. If a copper layer is present directly beneath the antenna, currents induced in the antenna will also induce a return current in the copper layer, but in the opposite direction. This generates an opposing E-field and H-field that destructively interfere with the initial E-field and H-field [27]. As a result, the radiated power will decrease, and the antenna may even stop radiating.

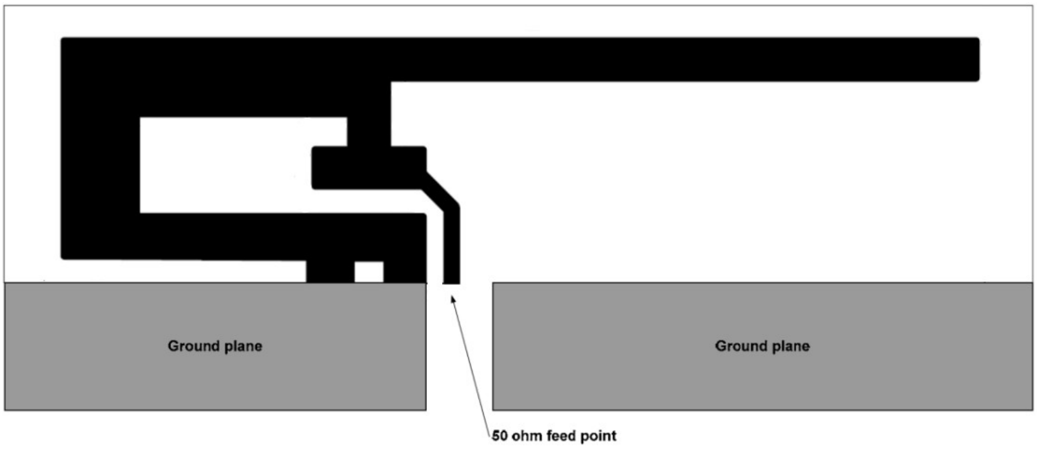


Figure 2.3: Inverted F-antenna [29]

2.2.7 Ripple Voltage and Ripple Current

Ripple voltage is the small amount of residual periodic variation of the DC output of a power supply that has been derived from an AC source[30]. This ripple is due to the incomplete suppression of the alternating waveform after rectification. Ripple voltage is more precisely defined as the RMS value of the AC component of the output voltage.

Ripple current refers to the AC component of the current drawn by a load connected to the DC output of a power supply[31]. In the context of capacitors in power supply units, ripple current is a crucial parameter as it produces heat within the capacitor, and excessive ripple current can lead to premature failure of the capacitor.

In an ideal power supply, the output voltage would be purely DC, and the output current would be purely constant. In reality, some AC component always remains in the output voltage and current due to factors such as load characteristics, inherent supply fluctuations, and circuit design limitations. This AC component is what we refer to as the ripple [32]. Figure 2.4 shows ripple build-up through a capacitor.

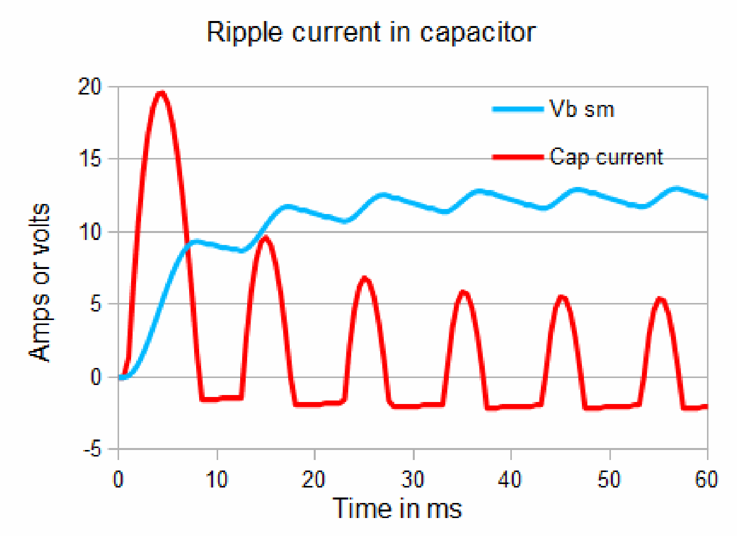


Figure 2.4: Ripple current [33]

2.2.8 Manhattan routing

Manhattan routing involves routing the traces in the X and Y directions, perpendicular to each other, with 90-degree bends at the corners. This routing style reduces the length of the traces and minimizes the number of vias required, resulting in a compact and efficient PCB layout [34]. Additionally, it simplifies the routing process, making it easier to plan and implement complex PCB designs.

One of the benefits of Manhattan routing is that it reduces EMI between traces. The perpendicular traces create a natural shield, preventing cross-talk and EMI between the traces. This routing style also allows for easy identification of the different signals on the PCB, as each signal follows a specific path along the X or Y direction.

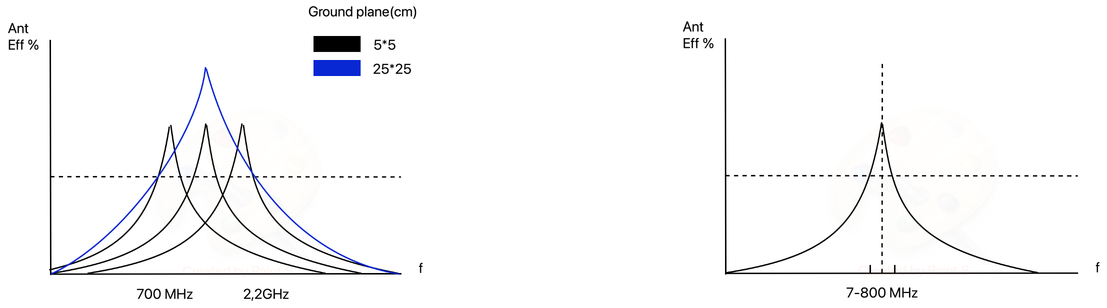
Sharp 90-degree bends in the traces can create corner reflections and increase the impedance of the traces[35], which can lead to signal integrity issues. Additionally, the routing style may only be suitable for some PCB designs, such as those with significant components or components that require specific placement.

2.2.9 RF Switches

RF switches are important for wireless communication systems, as they enable selective routing of radio frequency (RF) signals between different paths. The Nordic Semiconductor nRF9160 Development Kit enables antenna connection for LTE communication across multiple national frequencies. The kit is equipped with multiple RF switches that can dynamically change the frequency configuration of the antenna depending on the specific use case or environment [36]. The RF switches have low insertion loss and high isolation to ensure reliable and high-quality signal transmission and could operate at high frequencies.

Diagram 2.5a illustrates how RF switches are used on a smaller PCB to configure the antenna frequency [37]. In contrast, the larger blue graph PCB features a bigger ground plane and does not require RF switches to cover all necessary frequencies for the antenna to function effectively.

The diagrams are intended to demonstrate our approach to antenna design and illustrate the differences between using RF switches versus optimizing the antenna for a specific frequency range. The dashed lines shows the threshold for when the antenna is working. As a reference, the average antenna performance at 746-803 MHz for Thingy:91 is 12,6% [38].



(a) With RF switching

(b) Without RF switching

Figure 2.5: Antenna signal theory - RF Switching

2.3 Protocols

This section provides an overview of the different protocols used in the project, and explains their basic principles and purposes.

2.3.1 LTE Modem

The nRF9160 provides versatile options for wireless Machine-to-Machine communication. Its LTE modem supports both LTE-M and Narrowband IoT (NB-IoT), which outperform traditional LTE in terms of lower power consumption and improved coverage [39]. The modem employs eDRX and PSM, power-saving mechanisms specifically designed for NB-IoT and LTE-M, that enable IoT devices to conserve energy while maintaining network connectivity. Specifically, eDRX reduces the frequency of active listening, and PSM allows devices to enter deep sleep for whatever time is needed between transmissions. Figure 2.6 illustrates these mechanisms.

Together, these features enhance the nRF9160's capabilities in IoT applications by extending battery life and ensuring reliable connections [39].

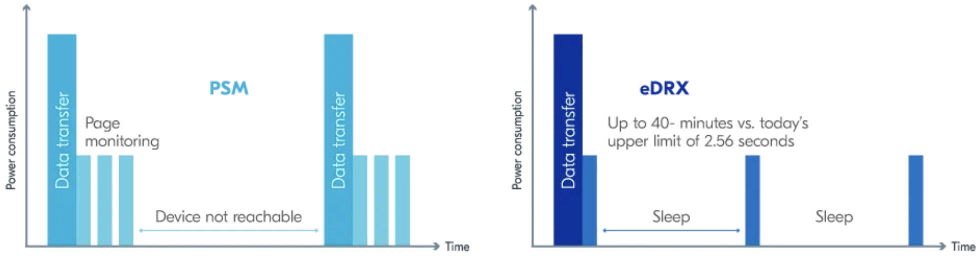


Figure 2.6: Difference between PSM and eDRX [40].

2.3.2 GNSS

The integrated GPS receiver within the nRF9160 is capable of supporting GPS L1 C/A and QZSS L1 C/A reception. The L1 C/A signal is a part of the GPS (Global Positioning System) operated by the United States. L1 refers to the frequency band centered at 1575.42 MHz, which is the primary civilian frequency used by GPS. C/A stands for "Coarse/Acquisition," representing a specific type of signal code that is designed for civilian use [41]. QZSS (Quasi-Zenith Satellite System) is a regional satellite navigation system operated by Japan, primarily aimed at improving the availability and accuracy of GPS signals in the Asia-Oceania region. QZSS L1 C/A refers to the signal transmitted by QZSS satellites on the same L1 frequency band as GPS and using the same C/A code [42]. This compatibility allows GNSS receivers, like the one in the nRF9160, to use QZSS signals in conjunction with GPS signals to enhance positioning performance. A key output of GNSS receivers is the PVT (Position, Velocity, and Time) estimate. This solution provides information on the user's position (latitude, longitude, and altitude), velocity (speed and direction), and time, all of which are derived from the signals received from multiple GNSS satellites.

Within this context, the term ”fix” refers to the successful determination of the receiver's precise geographic coordinates. This is achieved through the calculation of the time delays in the signals received from at least four satellites, which are then used to solve for the receiver's position in three dimensions and the time offset in the receiver's clock [43].

Further, the nRF9160 allows for three different operation modes [44]:

- **Single fix mode:** The GNSS receiver is active until a valid PVT estimate is produced, after which it is automatically switched off.
- **Continuous navigation mode:** The GNSS receiver is active continuously, producing PVT estimates at a 1 Hz rate.
- **Periodic navigation mode:** The GNSS receiver is turned on periodically to produce valid PVT estimates based on a specified fix interval (10-65535). The receiver is turned off after each valid PVT estimate.

2.3.3 Bluetooth LE

Our product requires a solution that facilitates reliable communication between devices while also being energy efficient. Bluetooth Low Energy (Bluetooth LE) fits these requirements, being a protocol designed for low-power data transmission, making it ideal for battery-driven applications such as IoT devices and wearable technology [45].

Operating band	2400 MHz – 2483.5 MHz ~2.4 GHz
Channel bandwidth	2 MHz
Number of RF channels	40
Maximum transmit power	20 dBm 0.1 W
Maximum application data throughput	1.4 Mbps
Maximum range at reduced data rates (125 & 500 kbps)	~1000 m

Table 2.1: Bluetooth LE characteristics [8]

The Bluetooth LE stack is divided into two sections: Host and Controller, which the application layer interacts with through APIs. The Host consists of protocols such as L2CAP, SMP, ATT, GATT, and GAP, while the Controller includes the Physical Layer (PHY) and Link Layer (LL) [46]. Figure 2.7 illustrates the layers making up the Bluetooth LE stack.

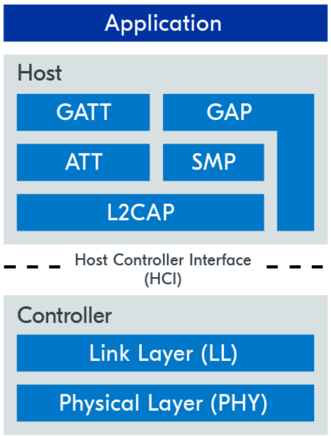


Figure 2.7: Bluetooth Low Energy stack layers [8].

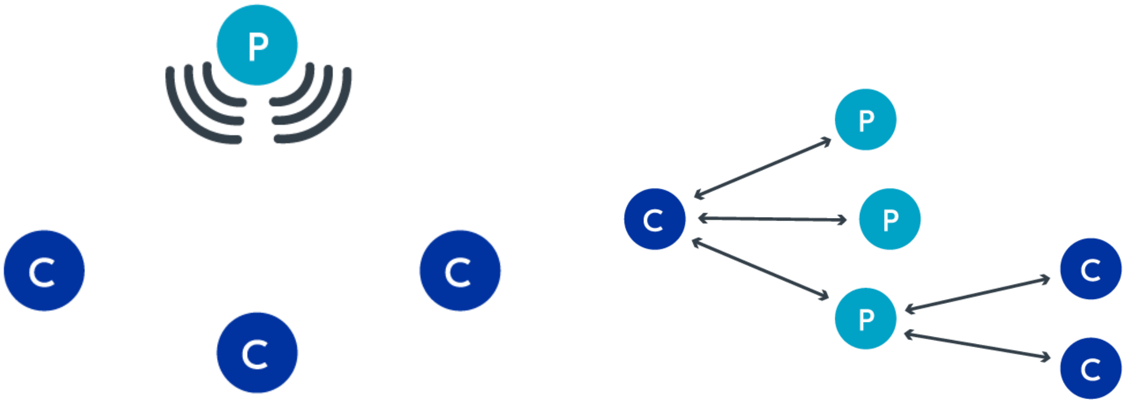
Bluetooth LE utilizes two network topologies defined in the GAP layer: broadcast and connected, as shown in Figure 2.8. Broadcast topology is power-efficient and suitable for unlimited devices, while connected topology offers higher throughput but has limitations due to bandwidth and hardware constraints. Devices can assume multiple roles, enabling a multi-role topology [47].

Bluetooth LE provides three types of Physical Layer (PHY) protocols: 1 Mbps, 2 Mbps, and coded PHY. Each utilizes a distinct modulation approach, which affects their respective data transmission speeds and range. Essentially, the PHY protocol chosen will dictate the balance between data transfer rate and communication distance [48].

Advertising is used to broadcast device presence and available services, while security features such as the Security Manager Protocol (SMP) and the Generic Attribute Profile (GATT) ensure secure communication between devices [49].

Devices in advertising mode, called peripherals or servers, continuously transmit advertising packets, which contain information such as device name, services offered, and connection parameters. These packets are received by nearby devices, called centrals or clients, in scanning mode, which can then initiate a connection with the advertising device. Advertising packets can also be used to transmit small amounts of data between devices without the need for a connection [49].

For added security, one can implement a pairing and bonding process. During pairing, devices exchange security keys and establish a shared secret key that is used to encrypt data during communication. The GATT layer provides additional security through the use of access permissions, which control who can read, write, or modify certain attributes exposed by a device [50]. This is often achieved through asking the user to press a button or to enter a pin code when connecting to a Bluetooth device.



(a) Broadcast topology

(b) Connected topology

Figure 2.8: Network topologies in Bluetooth LE

2.3.4 I2C

The Inter-Integrated Circuit (I2C) protocol is a widely used and straightforward communication protocol for connecting peripheral devices, such as sensors and microcontrollers, over short distances. It is a synchronous, serial, and bus-based protocol that requires only two bidirectional open-drain lines: the Serial Data Line (SDA) and the Serial Clock Line (SCL) [51]. This simplicity makes it suitable for resource-constrained applications.

The I2C protocol utilizes a master-slave architecture, with the master device generating the clock signal and initiating communication with slave devices on the bus. Each slave device has a unique address, which allows multiple devices to share the same communication lines. Data transfer takes place in byte-sized packets, with an acknowledge (ACK) or not acknowledge (NACK) signal sent after each byte to confirm successful data reception or indicate an issue. The majority of I2C controllers found on Nordic's chips are capable of supporting multiple speeds: 100 kbps , 400 kbps, and 1000 kbps. By default, the speed is set to 100 kbps [52].

2.3.5 SPI

The SPI is a high-speed, synchronous, serial protocol for connecting microcontrollers with peripheral devices. It uses a full-duplex, master-slave system with four signal lines: Serial Clock (SCK), Master Out Slave In (MOSI), Master In Slave Out (MISO), and Slave Select (SS). SPI is known for its speed, simplicity, and ability to handle multiple slaves[53].

The main differences between SPI and I2C are:

- SPI is faster than I2C.
- SPI uses four signal lines, while I2C only requires two.
- SPI supports full-duplex communication, while I2C is half-duplex.
- SPI does not use device addressing, while I2C requires unique addresses for each slave device.

2.3.6 UART

UART, which stands for Universal Asynchronous Receiver/Transmitter, is a hardware communication protocol used for serial communication between devices. It's a key component in many embedded systems and computers, enabling them to exchange data in the form of digital signals.

The *asynchronous* in its name refers to the fact that UART does not require a clock signal to synchronize the transmitting and receiving devices, unlike synchronous communication protocols. Instead, both devices must agree on a data rate, also known as baud rate, before communication starts.

UART communication involves two lines - a transmit line (TX) and a receive line (RX). Data is sent over these lines as a series of high and low voltages, representing binary 1s and 0s. The simplicity of UART, requiring minimal hardware, makes it a popular choice for simple, low-speed communication between devices [54].

2.4 Software

What follows is all the necessary information about the software the group utilized in this project.

2.4.1 Altium

Altium is an electronic design automation (EDA) software, enabling engineers to design and develop printed circuit boards (PCBs) for a wide range of applications. In our project, we used Altium to design the custom PCBs for the main and auxiliary units, incorporating the necessary components, sensors, and communication capabilities required for our system [55].

2.4.2 Ansys electrical

Ansys Electrical is a powerful simulation software used for various engineering applications, including electromagnetics, structural mechanics, and fluid dynamics. In our project, we used Ansys to simulate the performance of our system's RF components, such as antennas and RF switches, and to optimize their design for reliable wireless communication [56].

2.4.3 LTpowerCAD

LTpowerCAD is a power supply design tool from Analog Devices that assists engineers in selecting, designing, and optimizing power supplies for various applications. In our project, we used LTpowerCAD to design efficient and stable power supply solutions for our units, ensuring that they have sufficient power to operate reliably in a marine environment [57].

2.4.4 Github

Git is a distributed version control system that allows multiple developers to work on the same codebase simultaneously. GitHub is a web-based platform built on top of Git, providing a collaborative environment for project management and version control. Git and GitHub were both utilized throughout the project to manage the source code and collaborate on development [58]. The repository used for this project can be found here.

2.4.5 Zephyr RTOS and Devicetree

The Zephyr RTOS is an open-source, scalable, real-time operating system optimized for IoT devices [59]. It supports efficient multitasking via concurrent logic units called threads [60].

System configuration in Zephyr is governed by Kconfig and prj.conf.

Kconfig sets compile-time configuration options and generates a .config file containing selected configuration symbols and values [61]. It enables or disables specific features, influencing the system's behavior.

Conversely, prj.conf, located in the application's directory, is a project-specific .config file. It sets the values laid out in the Kconfig. [62].

Zephyr also uses Devicetree to describe hardware configurations without modifying the source code, facilitating firmware maintenance and updates across different devices [63]. Modifications are made via overlay files, altering only the relevant structures [64].

Zephyr also offers a wide array of APIs, making it easier for developers to start creating applications for different protocols without extensive knowledge of low level programming and timing. The Bluetooth API, for instance, offers functionalities such as device discovery, pairing, and data exchange with other Bluetooth devices. For bus communication protocols like SPI and I2C, Zephyr provides dedicated APIs that simplify data transfer between devices connected on the same bus. Beyond these, Zephyr also has integrated support for a wide range of sensors. APIs for these sensors help in the easy extraction and processing of sensor data, aiding in tasks like environmental monitoring, motion tracking, and more.

2.4.6 Visual Studio Code

Visual Studio Code, commonly referred to as VS Code, is a lightweight, open-source code editor developed by Microsoft that supports a wide range of programming languages and platforms [65]. It offers features such as syntax highlighting, code completion, refactoring, and debugging, which help developers write, test, and debug code more efficiently. VS Code also provides integration with Git, allowing developers to manage version control directly within the editor.

2.4.7 nRF Connect for Desktop

nRF Connect for Desktop is a cross-platform application suite designed for use with Nordic Semiconductor's devices and development tools [66]. It includes a variety of tools and utilities that assist developers in configuring, testing, and debugging their Nordic-based hardware and firmware projects. It also serves as a hub for installing and updating the SDK and VS Code extension.

2.4.8 nRF Connect for VS Code

nRF Connect for Visual Studio Code is an extension that makes it possible to work with Nordic Semiconductor's devices and development tools directly within the VS Code environment [67]. Extensions in VS Code are add-ons that enhance the functionality of the editor by providing additional features, tools, or support for specific programming languages and platforms. One useful feature is the ability to view the serial output coming from the development kits, directly inside VS Code by the click of a button, instead of having to rely on external terminal emulators like PuTTY.

Integrating Zephyr RTOS and nRF Connect

The nRF Connect SDK is built on top of the Zephyr RTOS[68], providing a scalable and modular platform for developing firmware for Nordic Semiconductor's devices. With nRF Connect for VS Code, we can take advantage of the features of both Zephyr RTOS and the nRF Connect SDK[7], using VS Code as their primary development environment. In the case of nRF Connect for VS Code, the extension focuses on streamlining the development process for Nordic Semiconductor's devices. It simplifies the tasks of setting up, configuring, and programming these devices.

2.4.9 Sketchup

SketchUp is a 3D modeling software widely used in architecture, engineering, and other designing fields[69]. It is tool used for visualizing and communicating design concepts. It offers tools for creating and modifying 3D models, with a simple 'push-pull' technique for easy shape manipulation. SketchUp provides a library of pre-made models, supports various export formats, and facilitates integration with other software.

2.4.10 FlashPrint

FlashPrint 5 is a 3D printing slicing software developed by FlashForge[70]. It offers advanced features for preparing and optimizing 3D models for FlashForge 3D printers. It allows users to import, manipulate, and customize models, generate support structures, and adjust other printing settings. FlashPrint 5 supports multiple file formats and provides a preview function for identifying potential issues before printing.

Chapter 3

Method

To demonstrate the desired functionalities for each unit and the order of implementation, three main steps have been outlined in table 3.1. The workflow progresses from top to bottom, ensuring completion of the current row before proceeding to the next. Parallel entries indicate that work should be conducted simultaneously. The first step, termed ”least viable product,” aims to create a functional standalone product, with subsequent steps, ”Vessel unit” and ”Auxiliary units,” focusing on integrating the other units to form a complete ecosystem.

The hardware and firmware design were developed concurrently, and all the initial coding and testing were done on the development kits, as explained in Section 3.2.

Units Development step	Main unit	Vessel unit	Auxiliary units
Step1: Least Viable Product	Connect to cloud, show position - LTE		
	Temperature and water sensor		
	Send information to cloud		
Step 2: Vessel unit		Connect to cloud, show position - LTE	
	Bluetooth signals to vessel unit	Bluetooth to main unit	
		Kill switch	
Step 3: Auxiliary units		Possibility to add and remove auxiliary units	Humidity sensor
			Send Bluetooth signals to vessel unit

Table 3.1: Development steps

3.1 Hardware design

When designing the hardware for this project, several critical factors were taken into consideration in order to achieve the best possible results. These factors were intricately balanced to ensure a successful prototype, while also minimizing costs and complexity. The primary considerations included:

- **Complexity:** The project had a limited timeframe for prototyping, design, and manufacturing. It was crucial to keep the systems as simple as possible to increase the likelihood of successful functionality on the first attempt and reduce the time spent on debugging. This was particularly important for wireless technology, which was considered above the project team's current level of expertise. Simplifying the design allowed for a more efficient development process and a smoother path to completion.
- **Size:** As the units were intended to be worn by users, minimizing their size was a top priority. This consideration, however, conflicted with the need for manufacturability. Striking a balance between the two was necessary to ensure that the final product was both compact and feasible to produce.

- **Cost:** Since the project was in its prototype phase, it was vital to keep costs to a minimum to avoid unnecessary expenditure. This required careful planning, resource allocation, and component selection to achieve the desired functionality without inflating the budget.
- **Manufacturability:** The PCBs required manual soldering, making manufacturability an essential factor when choosing and placing components. To maintain a balance between size and ease of assembly, most passive components were selected from the 0603 or 0402 series. These components were considered the smallest sizes that could still be debugged and fixed if necessary, but this choice did impact the overall size of the final product.

Thorough planning and meticulous attention to detail were necessary to ensure the best outcome for the project. By carefully balancing complexity, size, cost, and manufacturability, the project team was able to create a functional and efficient prototype that met the specific needs and constraints of the project.

3.1.1 Main unit

The main unit was the most sophisticated PCB designed during this project. To circumvent potential complications, the decision was made early on to model this after the Nordic Thingy:91 [4]. This approach offered several advantages, including:

- **Cost Reduction:** By drawing inspiration from the Nordic Thingy:91, the project team was able to leverage the existing resources and components already available from Nordic. This helped minimize the cost of the unit, making it more budget-friendly.
- **Accelerated Review Process:** Since the general design was familiar to Nordic, basing the main unit on the Nordic Thingy:91 allowed for a more efficient review process. This saved valuable time during the development stage, ensuring a more streamlined progression from design to implementation.

However, it is essential to note that the Nordic Thingy:91 is a development kit with numerous auxiliary features that were not necessary for the main unit's core functionality. To maintain simplicity and focus on the critical aspects of the project, many of these additional features were omitted from the design.

It was decided upon some key functionalities the design should include. A basic block diagram showcasing these features can be seen in Figure 3.1.

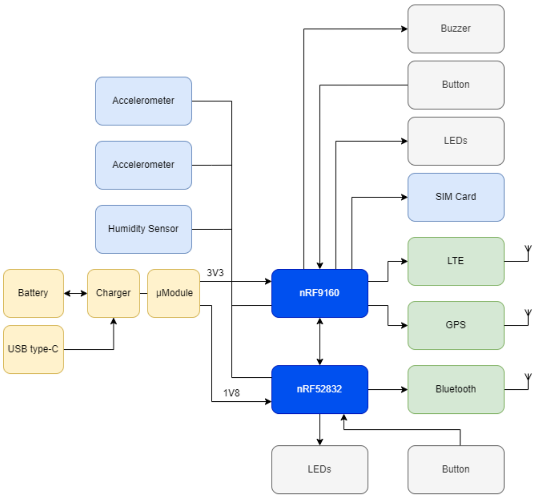


Figure 3.1: Main Unit design block diagram

The first prototype, shown in Figure 3.2 carried over some key features of the Nordic design. However, the later revisions proved to be quite different from the thingy:91.

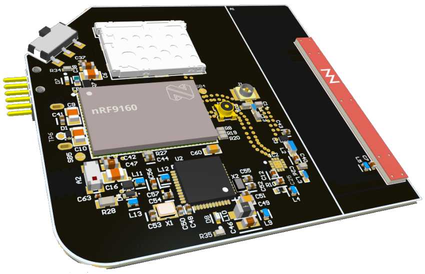


Figure 3.2: Early revision, from Altium Designer

Power supply

In order to determine the power source for the MOB (Mobile Unit), several criteria were considered, including voltage, current, and power requirements. Nordic had a readily available supply of 3.7V Li-PO batteries with a capacity of 1400mAh. Therefore, it was decided to base the initial design on this battery, with the flexibility to adjust its size if necessary during the development process.

To facilitate testing without the need for a battery, the design also included a USB Type-C connector. This connector enables the supply of 5V power to the DC-DC regulator, ensuring the functionality of the MOB unit during the testing and development stages.

With both the USB Type-C connector and the battery in place, the groundwork was laid for the integration of a battery charger. The inclusion of a battery charger would enhance the overall user experience of the MOB unit by allowing convenient and efficient charging of the battery.

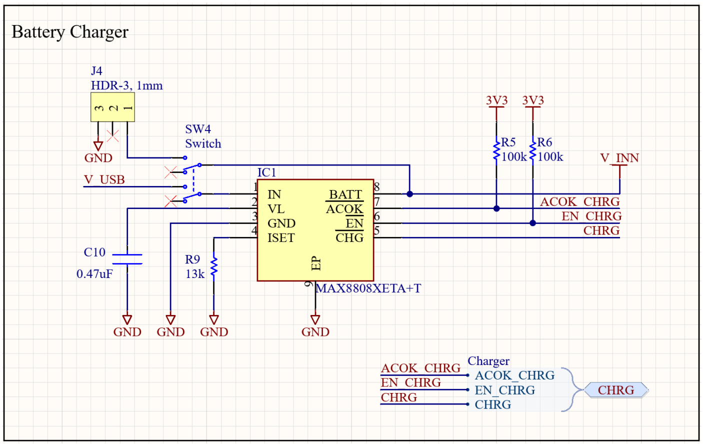


Figure 3.3: Battery charger schematics

A selection of DC-DC regulators were considered based on their specific features and capabilities. Three different regulators were compared: the LTM4622A, MAX774ESA+, and SR05S1V8.

The LTM4622A, manufactured by Linear Technology, was selected for its compact design, high power density, and wide input voltage range. This regulator integrates the necessary circuitry, inductor, and other components into a single package, providing a complete power solution. It offers adjustable output voltage, current limiting, and over-temperature protection, making it ideal for applications where space, efficiency, and reliability are crucial.

The MAX774ESA+ was included in the comparison as a voltage inverter or charge pump. This regulator converts a positive input voltage into a negative output voltage and is commonly used in applications such as LCD biasing or generating negative power supply rails. However, it has a fixed output voltage and may not offer the same level of versatility as the LTM4622A in terms of adjustable output voltage and additional features.

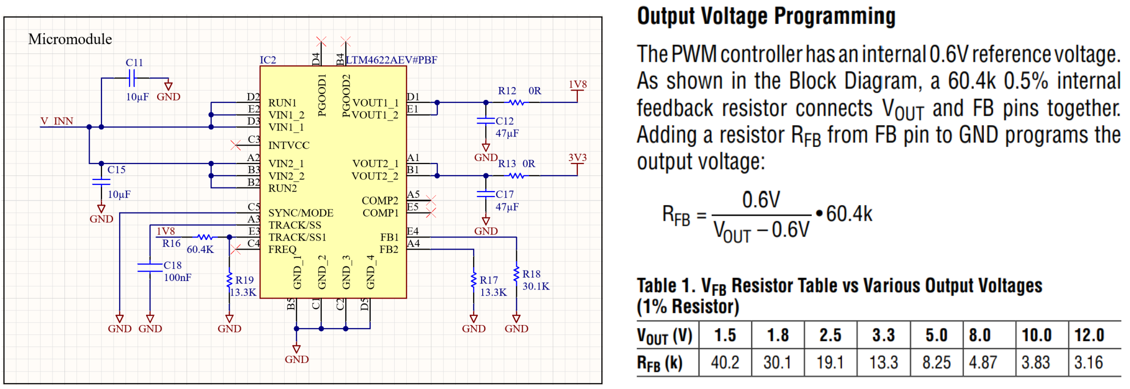
Lastly, the SR05S1V8, a voltage regulator, was also included in the evaluation. While specific details about this regulator are limited, it is likely a linear regulator or a simple DC-DC converter. Linear regulators generally have lower efficiency compared to switching regulators like the LTM4622A, and they may dissipate more power as heat.

	LTM4622A	MAX774ESA+	SR05S1V8
V_out [V]	1.5-12	3-36	1.8
Current	2A	100uA	500mA
N.O. outputs	2	1	1
Soldering	Difficult	Easy	Easy
Price	148kr	101kr	82kr
Package	25-LGA	8-SOIC	3-SIP Module

Table 3.2: Decision matrix for regulator

The final decision landed on the LTM4622A Micromodule. Most of the other solutions were either too large, required multiple regulators or were too expensive. The LTM4622A was a nice compromise, at the cost of being harder to solder than the other options due to its BGA package.

One of the key advantages of using the LTM4622A is its two individual voltage lines. This provides more flexibility and control over the power supply of the PCB, allowing for a more effective distribution of power. This is particularly important in small PCB designs where space is limited and efficient use of space is essential.



Output Voltage Programming

The PWM controller has an internal 0.6V reference voltage. As shown in the Block Diagram, a 60.4k 0.5% internal feedback resistor connects V_{OUT} and FB pins together. Adding a resistor R_{FB} from FB pin to GND programs the output voltage:

$$R_{FB} = \frac{0.6V}{V_{OUT} - 0.6V} \bullet 60.4k$$

Table 1. V_{FB} Resistor Table vs Various Output Voltages (1% Resistor)

V_{OUT} (V)	**1.5**	**1.8**	**2.5**	**3.3**	**5.0**	**8.0**	**10.0**	**12.0**
R_{FB} (k)	40.2	30.1	19.1	13.3	8.25	4.87	3.83	3.16

(a) Micromodule Schematics

(b) Voltage programming, from datasheet [71]

Figure 3.4: Micromodule implementation

The LTM4622A is set to deliver 1V8 and 3V3. This is done through the equations in figure 3.4b. The output lines are equipped with 0 Ω resistors, One for each of the voltage lines, and also one for each of the microcontrollers. The resistors can be soldered off, and wires can be soldered onto the pads for measuring current. Using header pins was considered for this, but they were avoided due to their larger size, and placement would prove to be difficult.

Humidity sensor

One of the key features of the main unit is the ability to detect if the user has fallen into water. To do this, a humidity sensor had to be implemented. A group of sensors were considered for this.

	SHT40	**BME680**	BME688
VDD [V]	1.08-3.6	1.7-3.6	1.7-3.6
Current [μA]	0.4	3.7 (Measure all channels)	3.9 (Measure all channels)
Measurement	Humidity	Humidity, Temperature, Gass	Humidity, Temperature, Gass
Package	Small	Small	Small
Price [NOK]	30	130	160
Protocol	I2C	I2C, SPI	I2C, SPI

Table 3.3: Decision matrix for humidity sensor

After evaluating the technical requirements and constraints of our system, we selected the BME680[72] sensor over the BME688[73] and SHT40[74] based on factors such as cost, power consumption, and functionality. Table 3.3 shows the decision matrix for humidity sensors

The BME680 consumes less power than the BME688, which is a critical consideration for our low-power device. Moreover, the BME688 is costlier, which would have increased the project's expenses. The longer range of the BME688 was not a crucial factor for the sensors implementation in our device.

We also ruled out the SHT40 sensor because it only measures humidity, while the BME680 can measure temperature, gas and humidity. This versatility enables us to collect more comprehensive data for our asset tracker application.

Accelerometers

To aid with the detection of whether or not the user of the Main Unit has fallen into water, accelerometers were to be added to the main unit. Again, a selection of sensors was considered for this.

	MC3419	**ADXL372**	**ADXL362**	IIM-42352
VDD [V]	1.7-3.6	1.6-3.5	1.6-3.5	1.71-3.6
Current [μA]	77	22	1.8	280
Measurement	X,Y,Z	X,Y,Z Temperature	X,Y,Z Temperature	X,Y,Z
Package	Small	Small	Small	Small
Price [NOK]	20	223	109	101
Protocol	I2C, SPI	SPI	SPI	I2C, I3C, SPI
Impact detection	No	Yes	No	Yes

Table 3.4: Decision matrix for accelerometer

In the end, it was decided to use two accelerometers, the ADXL372[75] and the ADXL362[76]. Both of these are good accelerometers but have their specific uses. The ADXL372 is a High-G accelerometer, whilst the ADXL362 is a low-power accelerometer. The ADXL372 has an operating range of up to ± 200g and current consumption of 22μA, while the ADXL362 has an operating range of up to ± 16g current consumption of 1.8μA. This allows for a wide range of accelerations to be measured, from small vibrations to larger shocks and impacts making them easily configurable in the main unit.

The noise levels of these accelerometers are as low as 175 $\mu g/\sqrt{\text{Hz}}$, making them ideal for detecting subtle vibrations and movements in your asset tracker. Furthermore, the ADXL362 and ADXL372 accelerometers are compact and small, which is essential for fitting them into a small IoT device.

By having these two work in unison, the ADXL362 could "wake up" the more accurate ADXL372. This solution would give the main unit low power consumption, as well as high accuracy.

These are also the sensors used on the Thingy:91. By using the same sensors as the Thingy:91 the development process is simplified. The sensors have been thoroughly tested and validated, reducing the amount of work required to ensure that they work correctly on the MOB sensor.

Additionally, the use of well-known sensors can make it easier to find resources and support when needed.

Microcontroller interface

One of the larger changes from the Nordic design was the swap from the nRF52840 to the nRF52832. The nRF52840 uses a QFN package, with a pitch of 0.35 mm, which is one of the smaller ones on the market. Such a small component would make soldering the PCBs by hand very difficult. Although the nRF52832 is by no means an easy package to solder, it is simpler to work with than the QFN on the nRF52840. This change does come with its downsides, like:

- Removal of USB lines
- Less GPIO pins
- Larger package
- Higher current consumption

Other comparisons can be found in table 3.5.

	nRF52840	**nRF52832**	nRF5340[77]
Supply Voltage [V]	1,7 – 5,5	1,7 – 3,6	1,7 – 5,5
Package	73-QFN (7x7)	48-QFN (6x6)	94-aQFN (7x7)
Pins	48	32	48
Dataline	Yes	No	Yes
Size	Small	Large	Small
Protocol	Bluetooth 5.0	Bluetooth v5.3	Bluetooth v5.1, Thread
Current consumption	6,3mA - 16.4mA	6,5mA - 71.mA	2,6mA – 7mA
Interface	I^{2}C, I^{2}S, SPI, UART, USB	I^{2}C, SPI, UART	ADC, I^{2}S, PWM, SPI, UART, USB

Table 3.5: Decision matrix for Bluetooth Chip

nRF52382 to nRF9160 via GPIO communication. This allows the two microcontrollers to communicate via the MCU_IF functionality.

USB

The early revisions of the Main Unit also featured USB communication to the Bluetooth chip. The Unit was then equipped with a Mini USB, and dual data lines were calculated to an impedance of 50Ω. When the Bluetooth chip was finally decided, it did not have the data line input originally planned, and this feature was removed. This topic is discussed further in Section 5.3.1.

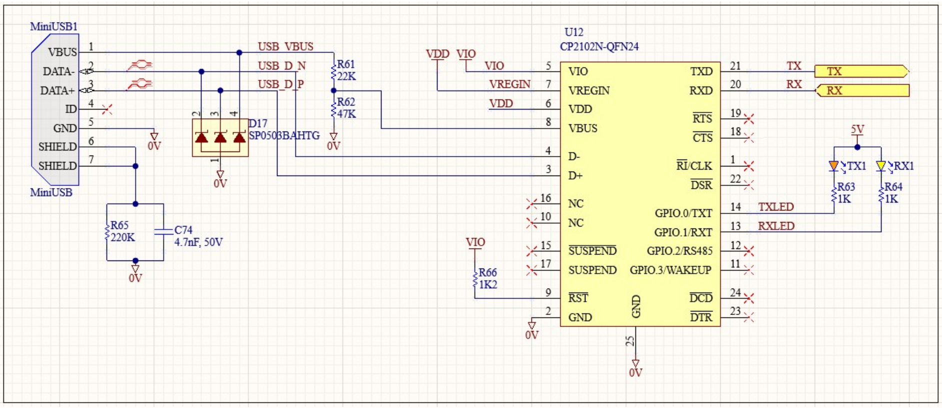


Figure 3.5: USB to UART translator. This was developed, but never implemented

LTE antenna

For antenna selection, Nordic Semiconductor has provided its own guidelines: nRF9160 Antenna and RF Interface Guidelines [78]. It is necessary to adhere to certain recommendations, as outlined in table 3.6. The choice of antenna should be made based on various parameters, including voltage standing wave ratio (VSWR), return loss, efficiency, maximum gain, and maximum input power.

Parameter	Requirement	Note
50 Ω impedance, antenna Voltage Standing Wave Ratio (VSWR)	<2:1	Recommendation
	<3:1	Absolute requirement
50 Ω impedance, antenna Return Loss (RL)	>9.5 dB	Recommendation
	>6.0 dB	Absolute requirement
Efficiency	>50%	Recommendation. Operators may have other efficiency requirements.
Maximum gain	[1]	
Maximum input power	1 W	

Table 3.6: Antenna requirements from Nordic [78]

The early designs used an Ignion trio M-Extend [79] antenna and RF switches. After feedback from Nordic, the group decided to switch the LTE antenna. The RF switches overcomplicated the system, and the prototype only needed to work in Norway. By using Ignions online antenna designer tool [80], the Ignion ONE mXTENDTM NN02-201 [81] was chosen. The NN02-201 is a compact, surface-mount antenna booster designed to enhance the performance of mobile devices, such as smartphones and tablets. It operates in the frequency range of 700MHz to 2700MHz, covering many of the popular cellular and Wi-Fi bands.

Technical features	**880 – 960 MHz**	**1710 – 2170 MHz**
Average Efficiency	>55%	>65%
Peak Gain	1.3 dBi	1.7 dBi
VSWR	<3:1	
Radiation Pattern	Omnidirectional	
Polarization	Linear	
Weight (approx.)	0.02 g.	
Temperature	-40 to +125 ºC	
Impedance	50 Ω	
Dimensions (L x W x H)	7.0 mm x 3.0 mm x 1.0 mm	

Table 3.7: Properties from Ignion Evaluation board[82]

The ignion NN02-201 also has an evaluation board. This was used as a reference during the design of the main unit.

RF switches for antenna connection

Initially, RF switching was discussed as a possible solution. However, after receiving feedback on the first revision of our PCB design, it was suggested that the RF switches be removed as they would require additional calibration time and could impact the size of the ground plane.

As a result, it was decided to focus on designing an antenna that would work for Norwegian antenna conditions, where the cellular frequency used by Telenor is typically around 800 MHz [83]. The antenna design was tailored to optimize performance within this frequency range, which achieved the desired level of performance without the need for RF switches 2.5b. This simplified the design process and reduced the complexity of the PCB layout, while still ensuring reliable communication over the desired frequency range.

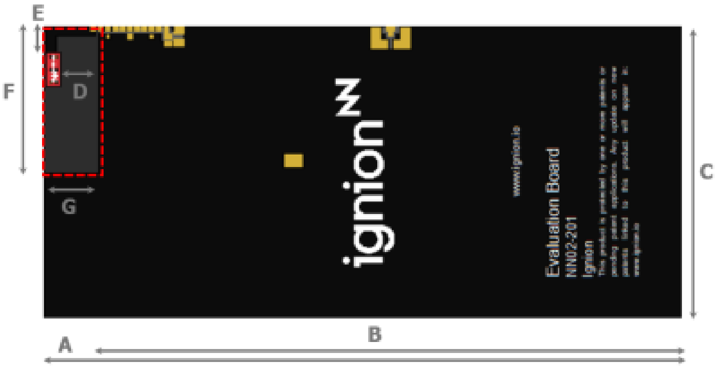


Figure 3.6: Ignion evaluation board [82]

Bluetooth Antenna

For Bluetooth communication, two options were considered: Making an integrated PIFA antenna or using a Simple 2.4Ghz chip antenna. Both of these options had their positives and negatives, as can be seen in table 3.8:

	Intigrated PIFA	Chip antenna
Band width	Limited	Large
Intigration	Hard	Easy
Size	Medium	Large
Cost	None / Low	10kr
Tunability	Limited	Good

Table 3.8: Comparison between chip and PIFA antenna

In the end, the group decided on using a PIFA antenna for the Auxiliary unit, and a chip antenna for the MOB unit. This would make the main unit easier to produce, and allow the auxiliary unit to be smaller.

For Bluetooth communication on the MOB unit, a Johanson Technology Inc. 2450AT18D0100001E [84] chip antenna was used. The 2450AT18D0100001E is a surface-mount ceramic chip antenna designed for use in the 2.4GHz band, commonly used for Bluetooth, Zigbee, Wi-Fi, and other wireless communication protocols. The antenna is small, measuring 1.6mm x 0.8mm x 0.55mm, and has a nominal impedance of 50 Ω. It offers a peak gain of 2.2dBi and an efficiency of 57%. This made the chip antenna compatible with the project's requirements. The antenna's datasheet states that it needs a clearance area of 4 x 6mm, as can be seen in figure 3.7

GNSS Antenna

An external GNSS antenna was employed for the main unit. This was because adding another antenna to the PCB proved too time-consuming. Consequently, the unit is outfitted with a UFL connector and facilitates power supply to the GPS antenna via an RF filter.
The decision landed on a Taoglass AGGBP.SL.18A active antenna [85]. This is a high-performance, compact antenna designed for use in GNSS applications. It supports GPS, GLONASS, Galileo, and BeiDou constellations, ensuring reliable and accurate positioning data across a wide range of devices and industries. With its 18x18mm form factor, the AGGBP.SL.18A offers a low-profile, space-saving solution for integration into various products. Its active LNA provides excellent signal amplification while maintaining low power consumption, making it suitable for battery-powered devices.

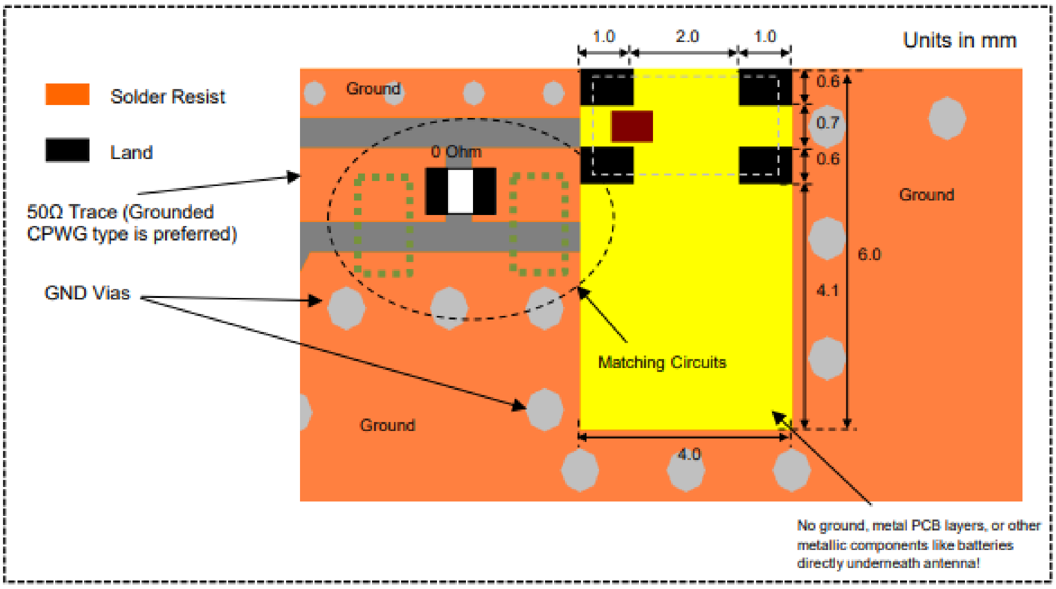

Units in mm
1.0
2.0
1.0
Solder Resist
Land
50Ω Trace (Grounded CPWG type is preferred)
GND Vias
Ground
0 Ohm
0.6
0.7
0.6
6.0
4.1
Ground
Matching Circuits
Ground
4.0
No ground, metal PCB layers, or other metallic components like batteries directly underneath antenna!

Figure 3.7: Bluetooth chip clearance area [84]

3.1.2 Auxiliary Unit

The Auxiliary unit is a much simpler system, designed with basic monitoring in mind. It is produced as a 4-layer PCB. This was done so that the main unit and auxiliary unit could be produced on the same production panel, reducing overall cost. The block diagram showcasing the major design features can be seen in Figure 3.8.

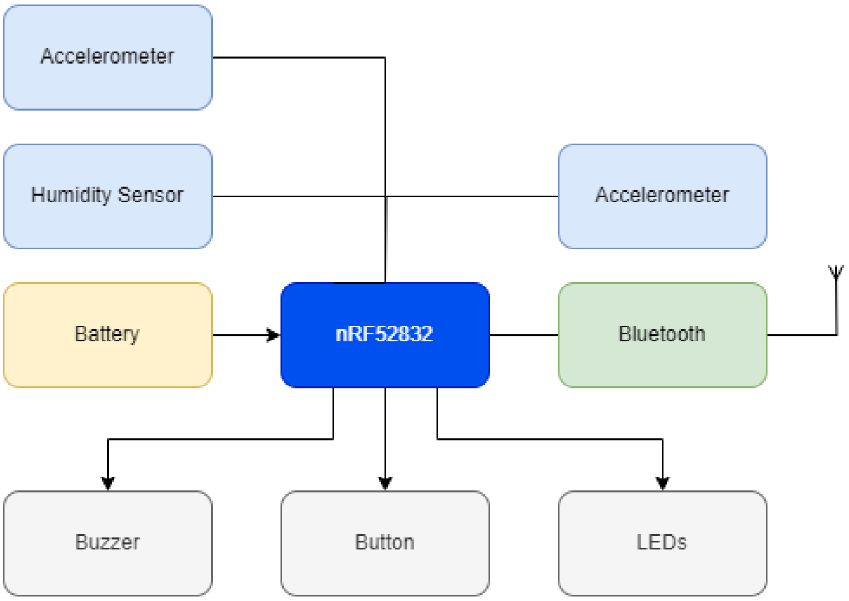


Figure 3.8: Auxilary Unit design block diagram

Using a PIFA antenna from Texas Instruments[29] simulated in Ansys Electrical[56], was decided early on. This would make the Unit small and easy to package. Via stitching is used to connect the different ground planes better. The vias also act as a barrier for the 2.4 GHz, protecting the rest of the circuit. This was incorporated after feedback from Nordic Semiconductors and draws inspiration from the nRF52DK layout[3].

The configuration of the nRF52832 SoC conforms to the layout recommendations outlined in its data sheet [86]. The high-frequency oscillator (X1) is stationed in the upper right corner, while the low-frequency oscillator (X2) is situated to the left of the nRF52832 [86]. To ensure optimal performance, the decoupling capacitors have been strategically placed as near as possible to their respective pins.

Power supply

To have the unit as small as possible, it was decided that the unit should run on a CR2032 battery. This battery has a nominal voltage of 3.3V, which is a satisfactory VSS of the nRF52832[86]. Seeing that the battery holder could easily be placed on the underside of the PCB, it was placed there.

Sensorics

The auxiliary unit and the Main unit utilize the same set of sensors. This was done in order to reduce the overall cost of the project, while also making it possible to re-use some of the software and layout of the different PCBs. The battery is connected to a switch, allowing the unit to be turned off if needed.

User Interface

To facilitate easy interaction with the unit and to simplify debugging, a comprehensive user interface was developed, as shown in Figure 3.9. The auxiliary unit, akin to the main unit, is equipped with two tactile buttons interfaced with the nRF52832 microcontroller, a piezoelectric buzzer, and an array of Light Emitting Diodes (LEDs).

One of the LEDs is connected to a GPIO line of the nRF52832, and the remaining one is connected to the voltage line powering the nRF52832. This dual connectivity provides two levels of user feedback. First, the LEDs confirm power delivery to the unit when illuminated, indicating that the device is active. Second, the LEDs' state—on, off, or blinking—can be manipulated through the GPIO line to represent more intricate feedback regarding the unit's operation. The programming flexibility of the GPIO lines makes this an effective and dynamic communication tool.

The buzzer serves as an audible alert system. It is designed to activate whenever the monitored values exceed predefined thresholds, thus warning the user of potential abnormal or critical conditions. The use of both visual (LEDs) and auditory (buzzer) feedback systems ensure that users are promptly and adequately informed about the unit's state, even in noisy or visually crowded environments.

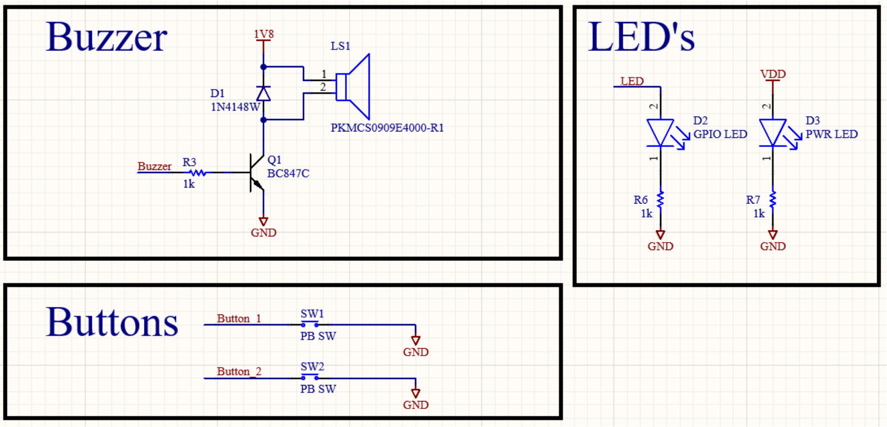
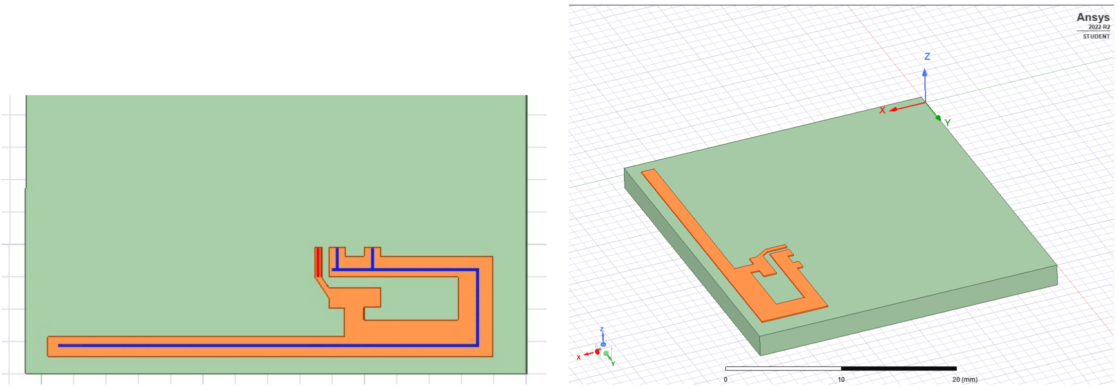


Figure 3.9: User interface schematics, Auxilliary unit

Antenna

The antenna used for the Auxilary Unit is a Texas Instruments PIFA antenna[29]. Due to suspicions that the impedance would be different from the one depicted in the design note [29], the antenna was simulated in Ansys Electrical [56]. By restricting the modification to the same variables showcased in the original design, mainly the feed port and overall length, adjustment of the antenna behavior became feasible. Figure 3.10b shows the modeled antenna in Ansys Electrical. The antenna was exported directly from Altium [55] before thickness and stackup were corrected.



(a) Top view

(b) 3D view

Figure 3.10: Antennas modeled in Ansys Electrical

With the correct stackup in Ansys electrical, it became clear that modifications were needed to meet the criteria of the original design [29]. As shown in Figure 3.10a, the thickness of the feed port, depicted in red in Figure 3.10a, has been altered to 0.432 mm to achieve a 50 Ω impedance. Additionally, the length of the antenna, depicted in blue, has been increased to 31.50 mm. These modifications resulted in an antenna with a 50 Ω feed port impedance and a quarter wavelength of 2.4 GHz when simulated in free space. The results are discussed further in Section 5.1.1.

3.1.3

3.2 Firmware implementation

The initial firmware development stages were done on the different development kits provided by Nordic Semiconductor, namely the nRF9160-DK, the Thingy:91, and the nRF52840-DK. By completing the Nordic Academy, the group members acquired the knowledge and necessary skills to begin development on these kits. Since our Vessel Unit is a Thingy:91, the firmware development on this unit was more straightforward than the other two units. We utilized the nRF9160 Development Kit for developing LTE and GPS code, and the nRF52840 DK for creating BLE code.

The group laid out a plan of how to implement the firmware, and what order it should be completed, as shown in figure 3.1. Substantial parts of the code is inspired by the courses introduced in Section 1.3.1 and form the samples included in the nRF SDK.

This is because the Main Unit and Auxiliary Unit are similar in functionality and sensor hardware, the difference lies in cellular communication capabilities. Therefore, instead of discussing each unit separately, we choose to introduce each firmware solution and concept collectively, highlighting the cohesive interplay within the whole system. In the subsequent sections, we will explore these firmware solutions, detailing their function in the system context.

We have structured the following sections to mirror the order in which the work was conducted.

3.2.1 Choice of RTOS Over Bare-metal Programming

In deciding between bare-metal programming and a Real-Time Operating System (RTOS), we chose to utilize the Zephyr RTOS, which is already utilized by Nordic Semiconductor. This decision was based on several key factors. Zephyr RTOS simplifies development by abstracting low-level hardware details and offering a high-level API for system functions. Its priority-based, preemptive scheduler ensures efficient task management, allowing for the handling of multiple concurrent tasks, such as sensor data acquisition and communication.

While bare-metal programming may offer improved battery life and energy efficiency, the advantages provided by an RTOS like Zephyr outweigh these benefits. RTOS-based designs enable easy expansion and modification, which is crucial for the long-term viability of the asset tracker (Section 5.3). The difference is shown in figure 3.11.

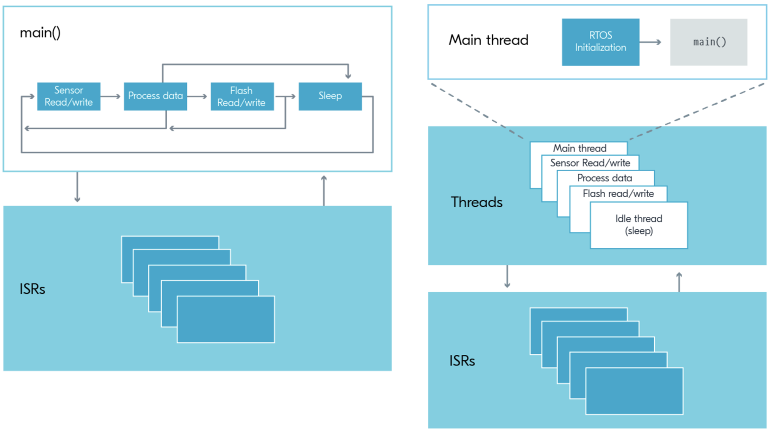


Figure 3.11: Visualizing the difference between bare-metal (left) and RTOS (right) [60]

3.2.2 Communication protocol

CoAP and MQTT are two popular protocols for IoT devices [87], but they have different characteristics that make them better suited for different use cases. CoAP, or Constrained Application Protocol, is a lightweight protocol designed for constrained devices and networks. Unlike TCP-based protocols like MQTT, CoAP uses UDP and is connection- less, meaning two endpoints can communicate without a prior arrangement. This makes CoAP a good choice for low-power devices that may not have a stable network connection or need to conserve battery life.

On the other hand, MQTT is a reliable, message-oriented protocol that uses TCP to establish a connection between two endpoints. This makes it a good choice for applications that require a reliable and persistent connection, such as real-time monitoring or control systems.

When it comes to transmitting GPS data from a boat, the use of CoAP is advantageous due to its lightweight nature. Additionally, the connection-less nature of CoAP can be beneficial when dealing with intermittent network connections, as it allows for more flexibility in how and when data is transmitted.

3.2.3 LTE-M

In developing the cellular communication we evaluated the suitability of both LTE-M and NB-IoT technologies, which are supported by the nRF9160. The selection process involved assessing their respective characteristics and how they align with the requirements of our project [40].

In the end, we chose LTE-M, considering several factors:

- Mobility Support
- Lower Latency
- Higher Data Rate

While NB-IoT has merits in power consumption and network coverage, LTE-M's features better cater to the specific requirements of our MOB accident tracking system, providing a more effective and reliable solution. Figure 3.12 shows the initialization of LTE-M in prj.conf.

```
CONFIG_LTE_NETWORK_MODE_LTE_M_GPS=y
```

Figure 3.12: Enabling LTE-M API

lte.c code on the Main Unit

Figure 3.13 presents an excerpt of the code implemented in the lte.c file. In the initialization phase, we defined a semaphore `lte_connected`, which was utilized in the main.c file to manage synchronization between different threads concerning LTE connectivity.

We used a function `lte_handler()` to handle LTE events and check the network registration status in the `LTE_LC_EVT_NW_REG_STATUS` case. Additionally, we utilized the function `modem_configure()` to initialize and connect to the LTE network, and deactivate LTE to enable GNSS. This switching between LTE and GNSS is necessary because the nRF9160 SiP's antenna shares the same RF resources for both GNSS and LTE-M/NB-IoT connectivity. It's not possible to operate both functionalities simultaneously.

```
#include "lte.h"
#include <zephyr/kernel.h>
#include <modem/lte_lc.h>
...
LOG_MODULE_REGISTER(LTE, LOG_LEVEL_INF);

K_SEM_DEFINE(lte_connected, 0, 1);

static void lte_handler(const struct lte_lc_evt *const evt)
{
        switch (evt->type)
        {
        case LTE_LC_EVT_NW_REG_STATUS:
                if ((evt->nw_reg_status != LTE_LC_NW_REG_REGISTERED_HOME) &&
                        (evt->nw_reg_status != LTE_LC_NW_REG_REGISTERED_ROAMING))
                {
                        break;
                }
                ...
                k_sem_give(&lte_connected);
                break;

        case LTE_LC_EVT_PSM_UPDATE:
                LOG_INF("PSM parameter update: TAU: %d, Active time: %d\n",
                                evt->psm_cfg.tau, evt->psm_cfg.active_time);
                break;
        case LTE_LC_EVT_EDRX_UPDATE:
        {
                char log_buf[60];
                ssize_t len;

                len = snprintf(log_buf, sizeof(log_buf),
                                           "eDRX parameter update: eDRX: %f, PTW: %f\n",
                                           evt->edrx_cfg.edrx, evt->edrx_cfg.ptw);
                ...
                break;
        }
        case LTE_LC_EVT_RRC_UPDATE:
                LOG_INF("RRC mode: %s\n",
                                evt->rrc_mode == LTE_LC_RRC_MODE_CONNECTED ?
                                                    "Connected" : "Idle\n");
                break;
        case LTE_LC_EVT_CELL_UPDATE:
                LOG_INF("LTE cell changed: Cell ID: %d, Tracking area: %d\n",
                                evt->cell.id, evt->cell.tac);
                break;
        default:
                break;
        }
}

void modem_configure(void)
{
        err = lte_lc_init_and_connect_async(lte_handler);

        /* Decativate LTE and enable GNSS. */
        if (lte_lc_func_mode_set(LTE_LC_FUNC_MODE_DEACTIVATE_LTE) != 0)
        {
                ...
        }
}
```

Figure 3.13: LTE.c

LTE code in main.c on the Main Unit

In the main.c file (Figure 3.14), we used the modem key management system to write an identity credential and a pre-shared key (PSK) to the modem in the `main()` function. A separate upload thread is spawned, handling the activation and deactivation of the LTE functionality, as part of the task scheduling capabilities provided by the Zephyr Real-Time Operating System. This thread operates alongside a semaphore mechanism, `sem_lte_busy`, which is utilized to ensure thread-safe access to the LTE functionality, thus promoting the reliable operation of the device's LTE communication processes.

```c
#include <modem/modem_key_mgmt.h>
#include "lte.h"

LOG_MODULE_REGISTER(main_c_, LOG_LEVEL_DBG);

#define UPLOAD_THREAD_PRIORITY 8
#define STACKSIZE 2048

K_THREAD_STACK_DEFINE(upload_thread_stack, STACKSIZE);
struct k_thread upload_thread_data;

K_SEM_DEFINE(sem_lte_busy, 1, 1);

void upload_thread(void *arg1, void *arg2, void *arg3)
{
    int err;
    ...
    err = lte_lc_func_mode_set(LTE_LC_FUNC_MODE_NORMAL);
    if (err != 0)
    {
        LOG_ERR("Failed to activate LTE");
        return;
    }
    ...
    err = lte_lc_func_mode_set(LTE_LC_FUNC_MODE_DEACTIVATE_LTE);
    if (err != 0)
    {
        LOG_ERR("Failed to deactivate LTE and enable GNSS functional mode");
        return;
    }
    k_sem_give(&sem_lte_busy); // Release the LTE semaphore
}

void main(void)
{
    int err;
    ...
    err = modem_key_mgmt_write(SEC_TAG, MODEM_KEY_MGMT_CRED_TYPE_IDENTITY,
                CONFIG_COAP_DEVICE_NAME, strlen(CONFIG_COAP_DEVICE_NAME));
    if (err)
    {
        ...
    }
    err = modem_key_mgmt_write(SEC_TAG, MODEM_KEY_MGMT_CRED_TYPE_PSK,
                CONFIG_COAP_SERVER_PSK, strlen(CONFIG_COAP_SERVER_PSK));
    ...

    modem_configure();
    k_sem_give(&sem_lte_busy); // Release the LTE semaphore

    k_thread_create(&upload_thread_data, upload_thread_stack,
                    STACKSIZE, upload_thread, NULL, NULL, NULL,
                    UPLOAD_THREAD_PRIORITY, 0, K_NO_WAIT);
}
```

Figure 3.14: LTE configuration in main.c

LTE code for the Vessel Unit

The Vessel Unit in LTE code structure mirrors the Main Unit. It has two responsibilities: downloading sensor data from the Tracker Units and uploading its own positional data. These tasks are handled by separate threads, download_thread and upload_thread. Threads coordinate using the sem_lte_busy semaphore to efficiently utilize the LTE connection. Threads wait for sem_send_new_data and sem_get_new_data semaphores to upload and download data, respectively. The LTE connection is deactivated after each operation to conserve power. Releasing the semaphore is crucial to prevent indefinite blocking.

3.2.4 GNSS Implementation

gnss.c

In the creation of the GNSS module (Figure 3.15), we constructed the required header file for GNSS operations, drawing upon the sample code found within the nRF SDK. The custom header file, named "gnss.h", was included using the directive *#include "gnss.h"*.

For thread synchronization during a successful GNSS fix, a semaphore named gnss_fix_sem was established. This was derived from our own manipulations of the SDK's sample code and was defined using K_SEM_DEFINE(gnss_fix_sem, 0, 1). This allowed the various threads to operate in harmony during the GNSS fix process.

A callback function, gnss_event_handler(), was created to run on a separate thread dedicated to GNSS operations. This function was designed to manage a range of GNSS events. The NRF_MODEM_GNSS_EVT_SLEEP_AFTER_FIX event, for example, let's the program know that a valid GPS position has been achieved and that data can be uploaded.

When the NRF_MODEM_GNSS_EVT_SLEEP_AFTER_FIX event is triggered, indicating a successful GNSS fix, the function was designed to read the PVT data from the sensor into a variable, and display the fix data. Simultaneously, it releases the gnss_fix_sem semaphore, signaling to other threads that a GNSS fix has been achieved and data can be transmitted.

The function gnss_init_and_start() was developed to initialize and activate the GNSS system. This function sets various parameters, including the timing source, use case, event handler, fix interval, and fix retry, before starting the GNSS. The GNSS priority mode was also enabled to ensure that the GNSS receiver would be prioritized when both the GNSS receiver and the LTE modem are needed, as they cannot operate concurrently. There's also a function for creating fake GPS data, for testing purposes.

```
...
#include "gnss.h"
#include <nrf_modem_gnss.h>
#include <zephyr/logging/log.h>
...
LOG_MODULE_REGISTER(GNSS, LOG_LEVEL_INF);
...
K_SEM_DEFINE(sem_send_new_data, 0, 1);
...
static void gnss_event_handler(int event)
{
    switch (event)
        {
        ...
        case NRF_MODEM_GNSS_EVT_SLEEP_AFTER_FIX:
                retval = nrf_modem_gnss_read(&last_pvt, sizeof(last_pvt),
                                                NRF_MODEM_GNSS_DATA_PVT);
                if (retval == 0)
                {
                        current_pvt = last_pvt;
                        print_fix_data(&current_pvt);
                        k_sem_give(&sem_send_new_data);
                }
                break;
        ...
}
...
int gnss_init_and_start(void)
{
...
        if (nrf_modem_gnss_timing_source_set(NRF_MODEM_GNSS_TIMING_SOURCE_TCXO))
        {
                LOG_ERR("Failed to set TCXO timing source");
                return -1;
        }
...
        if (nrf_modem_gnss_event_handler_set(gnss_event_handler) != 0)
        {
                LOG_ERR("Failed to set GNSS event handler");
                return -1;
        }
...
        if (nrf_modem_gnss_start() != 0)
        {
                LOG_ERR("Failed to start GNSS");
                return -1;
        }
...
        return 0;
}
```

Figure 3.15: gnss.c on the Main Unit

GNSS in main.c

As seen in Figure 3.16, the main.c code was developed with several global variables and semaphores defined. The `main()` function, serving as the program's entry point, was designed to initialize various components, including the modem, buttons, LEDs, and GNSS. Following these initializations, the program was set to enter a loop, where it waits for the `gnss_fix_sem` semaphore. Once the GNSS fix is obtained, the program activates the LTE modem, connects to the server, and sends the acquired position data.

```c
#include <zephyr/kernel.h>
#include <modem/modem_key_mgmt.h>
...
#include "gnss.h"
...
K_SEM_DEFINE(gnss_fix_sem, 0, 1);
...

void main(void)
{
    ...
    modem_configure();
        LOG_INF("Starting GNSS....");
        err = gnss_init_and_start();
        if (!err)
        {
                LOG_INF("GNSS started successfully\n");
        }
    k_sem_give(&sem_lte_busy); // Release the LTE semaphore after GNSS is setup
}
```

Figure 3.16: Relevant code in main.c for GNSS operation for the Main Unit

This pattern of alternating between GNSS and LTE is a classic approach for power optimization in IoT devices. Activating the LTE modem only when necessary (i.e., when there is data to be sent) can save a significant amount of power. Similarly, deactivating the LTE modem when not in use also helps to reduce power consumption. This is especially relevant in battery-powered IoT devices where optimizing power consumption is of importance.

3.2.5 Connecting to a server

Our project required establishing a connection to the Eclipse Californium public CoAP sandbox server [88]. This choice was influenced by its presentation in the Nordic Semiconductor's Academy Course, leading to its selection for our device. While this is a sandbox meant for testing purposes, it was deemed out of the scope of this project to setup our own CoAP server. The documentation for how to properly do so was also few and far between.

We used prj.conf to alter the Kconfig which holds the necessary connection settings. The name of our device, represented by the `COAP_DEVICE_NAME`, was defined here. Similarly, the hostname of the CoAP server was established using the `COAP_SERVER_HOSTNAME` directive and was set as `californium.eclipseprojects.io`. The port the CoAP server listens on was defined with `COAP_SERVER_PORT`, having a default value of 5684.

```
...
config COAP_DEVICE_NAME
        string "Device resource name - this will be the device name on the CoAP server"
        default "cali.MainUnit"
...
config COAP_SERVER_HOSTNAME
        string "CoAP server hostname"
        default "californium.eclipseprojects.io"

config COAP_SERVER_PORT
        int "CoAP server port"
        default 5684
...
```

Figure 3.17: Kconfig code to connect to a CoAP server

main.c

CoAP functionality was implemented in the coap.c and coap.h files, as to keep the codebase somewhat modular and easy to follow Figure 3.18. The `#include "coap.h"` directive was used to incorporate the CoAP library into our main.c file, from where we called on the functions to send and receive values form the CoAP server.

We set the maximum lengths for both the received and sent CoAP messages through the `APP_COAP_MAX_MSG_LEN` and `APP_COAP_SEND_MAX_MSG_LEN` directives, which were defined as 1280 and 128 bytes, respectively.

We declared two buffer arrays, `coap_buf[APP_COAP_MAX_MSG_LEN]` and `coap_databuf[APP_COAP_SEND_MAX_MSG_LEN]`, for storing the inbound and outbound CoAP messages.

The `server_connect(sock)` function was employed to establish a connection to the CoAP server. In the event of a failed connection attempt, an error message was logged and the program terminated.

The `client_post_send()` function, illustrated in Figure 3.18, sent a CoAP POST request to the server. This function utilized the socket connection, the send and receive buffers, and the current and last GPS coordinates. If the send operation was unsuccessful, the program logged an error message and broke from the loop.

Finally, the response from the server was handled by the `client_handle_get_response()` function. If the server's response was invalid, the program logged an error message and broke from the loop.

```c
...
#include "coap.h"
...
#define APP_COAP_MAX_MSG_LEN 1280
#define APP_COAP_SEND_MAX_MSG_LEN 128

static uint8_t coap_buf[APP_COAP_MAX_MSG_LEN];
static uint8_t coap_databuf[APP_COAP_SEND_MAX_MSG_LEN];
...
if (server_connect(sock) != 0)
{
    LOG_ERR("Failed to initialize CoAP client\n");
    return;
}

if (client_post_send(sock, coap_buf, sizeof(coap_buf), coap_databuf, sizeof(coap_databuf),
                     current_pvt, mob_event) != 0)
{
    LOG_ERR("Failed to send GET request, exit...\n");
    break;
}
received = recv(sock, coap_buf, sizeof(coap_buf), 0);
...
// Read it back
err = client_handle_get_response(coap_buf, received);
if (err < 0)
{
    LOG_ERR("Invalid response, exit...\n");
    break;
}
```

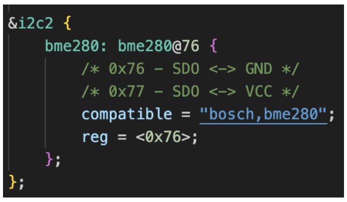
Figure 3.18: Code for CoAP in main.c on the Main Unit

3.2.6 Custom sensor library

The development of the Inter-Integrated Circuit (I2C) protocol was initially deployed the BME280 sensor. The selection of this sensor was based on its availability during the development phase and its integrated measurement capabilities: pressure, humidity, and temperature.

To ensure easy access to all of our sensors, we created a custom library, `sensors.h/sensors.c`. It consists of an organized structure for each sensor type, with a comprehensive parent structure to encapsulate all sensors' data collectively. The individual structures for the BME280, ADXL362, and ADXL372 sensors, namely `bme280_data`, `adxl362_data`, and `adxl372_data`, hold device pointers and sensor values related to their respective sensor types.

First, the devices are added to the devicetree overlay, as can be seen in Figure 3.19, where the BME280, which has an I2C address of 0x76 is added. Zephyr know that this is a BME280 temperature, humidity and pressure sensor because of the `compatible=vendor,model` characteristic, which it checks up against its built-in database of sensors.



```
&i2c2 {
    bme280: bme280@76 {
        /* 0x76 - SDO <-> GND */
        /* 0x77 - SDO <-> VCC */
        compatible = "bosch,bme280";
        reg = <0x76>;
    };
};
```

Figure 3.19: Devicetree overlay file f

After this, the device can be added to our custom library. For initialization and data acquisition, the library offers the `sensors_init(sensors_s *sensors)` and `read_sensors(sensors_s *sensors)` functions, respectively. The `sensors_init` function retrieves the devices from the devicetree and checks their readiness by calling `device_is_ready`. If a sensor device is not ready, a log message will be generated, providing diagnostic information.

The `read_sensors` function, on the other hand, fetches the data from each sensor by using the respective `readBME280`, `readADXL362`, or `readADXL372` functions. Each of these functions fetches the sensor sample and then gets the respective sensor channel data like temperature, pressure, humidity, or acceleration, storing the data in the appropriate sensor data structure.

This methodology forms an abstraction layer for managing the various sensors and allows for an efficient and straightforward approach to fetch and read sensor data, thus enabling us to concentrate on interpreting the data, rather than managing the data acquisition. Moreover, with Zephyr's integrated logging functionality, it becomes easier to identify and debug potential sensor-related issues.

The process of reading sensor data and printing the data from the BME280 sensor is shown in Figure 3.20. In this example, `read_sensors(&sensors)` is utilized, where `&sensors` denotes the memory address of the struct which houses our three sensors' data. The struct's memory address is passed as a reference to the function. Following the reading operation, the `.val1` and `.val2` members of the `sensor_value` structure are accessed to print the BME280 sensor data. These members of the `sensor_value` struct represent the integer and fractional parts of the sensor data, respectively.

```
void sensor_thread(void *arg1, void *arg2, void *arg3)
{
...
 while (1)
 {
        ...
    read_sensors(&sensors);

      // Will print sensor data if debug log level is enabled
      LOG_DBG("BME280: Temperature: %d.%06d C,Pressure: %d.%06d hPa,Humidity: %d.%06d %%RH\n",
            sensors.bme280.temperature.val1, sensors.bme280.temperature.val2,
            sensors.bme280.pressure.val1, sensors.bme280.pressure.val2,
            sensors.bme280.humidity.val1, sensors.bme280.humidity.val2);

    if (sensors.bme280.humidity.val1 > 60 && mob_event == false)
    {
        LOG_WRN("MOB ALERT!");
        ...
    }
    k_sleep(K_SECONDS(3));
 }
}
```

Figure 3.20: Code showing how to update the sensor data, in addition to how sensor data can be accessed and printed. If the integer part of the humidity value is over 60, it will sound the alarm.

3.2.7 Bluetooth LE

Bluetooth LE, developed on the nrf52840DK, is used for internal communication between the units in our system. This communication is facilitated through a custom service called GSS, short for GPS Sensor Service. The primary purpose of GSS is to ensure a modular approach to our design, with each attribute offered by the service being identifiable through UUID. The primary objective behind the incorporation of BLE was to enable the tracker units to periodically broadcast data to the Vessel Unit, without the need of a cellular connection. Even without coverage, this would alert the crew onboard the vessel of an incident through a local connection. By continually transmitting sensor data, the system is designed to rapidly detect if the individual wearing the device falls into water or loses connection, thereby functioning as a kill switch. This ensures safety to those working at sea, by providing prompt detection, as well as redundancy when used in combination with other safety mechanisms. By leveraging the principles of modular design, the ensuing code has been made functional for both tracker units.

Advertising Parameters

The BLE implementation begins with the setup of advertising parameters, as shown in Figure 3.21 below. These parameters dictate the behavior of the tracker units during the advertising phase, deciding whether or not the peripheral is connectable or a beacon, setting up the advertising interval, etc. The below parameters were introduced in [89].

```
static struct bt_le_adv_param *adv_param = BT_LE_ADV_PARAM((BT_LE_ADV_OPT_CONNECTABLE |
                                        BT_LE_ADV_OPT_USE_IDENTITY),
                                        800,  /*Min Advertising Interval 500ms (800*0.625ms)*/
                                        2400, /*Max Advertising Interval 1500ms (2400*0.625ms)*/
                                        NULL);/* Set to NULL for undirected advertising*/
```

Figure 3.21: Bluetooth parameters

Broadcast Data Structure

Figure 3.22 below illustrates the data to be broadcast by the tracker units during the advertising phase. The structure includes flags indicating the discoverability mode of the device, its lack of

BR/EDR support, the UUID of the Device Information Service (DIS) and some of its attributes.

```
static const struct bt_data ad[] = {
        BT_DATA_BYTES(BT_DATA_FLAGS, (BT_LE_AD_GENERAL | BT_LE_AD_NO_BREDR)),
        BT_DATA_BYTES(BT_DATA_UUID16_ALL, BT_UUID_DIS_VAL),
        BT_DATA(BT_DATA_MANUFACTURER_DATA, CONFIG_BT_DIS_SERIAL_NUMBER_STR,
        (CONFIG_BT_DIS_SERIAL_NUMBER_STR) - 1),
};
```

Figure 3.22: BLE - structure of data to be advertised

Scan Response Data

In response to a scanning request from another BLE device, in our case the Vessel Unit, the tracker units respond with scan response data, as shown in Figure 3.23 below. This includes the Universally Unique Identifier (UUID) for our custom GSS service, as well as some device information to help identify the device.

```
static const struct bt_data sd[] = {
        BT_DATA_BYTES(BT_DATA_UUID128_ALL, BT_UUID_GSS_VAL),
        BT_DATA(BT_DATA_NAME_COMPLETE, CONFIG_BT_DEVICE_NAME, sizeof(CONFIG_BT_DEVICE_NAME) - 1),
};
```

Figure 3.23: Scan response data

Connection Callbacks

Upon initialization, connection callbacks are registered and advertising is started. These callbacks are invoked when a connection is established or disconnected, as shown in Figure 3.24 below.

```
int ble_init(struct bt_conn_cb *connection_cb)
{
    ...
}
void on_connected(struct bt_conn *conn, uint8_t err)
{
    ...
}
void on_disconnected(struct bt_conn *conn, uint8_t reason)
{
    ...
}
```

Figure 3.24: Initialization and callback functions.

GSS Implementation

In addition to the implementation of BLE, our project also involved the development of a custom GATT service called GSS. This service was designed to convey the most important information from our tracker unit to the vessel unit, The structure of these attributes is detailed in the GSS Attribute Table, as shown in Table 3.9 below.

Service (GSS)	Handle	UUID	Attribute Permissions	Attribute Value
GSS Service Declaration	0x0001	0x2800	Read	6dea7370-0bdf-43db-8ea8-1c3a6a7e5909
GPS Charcteristic Declaration	0x0002	0x2803	Read	Properties: Read or indicate Handle of value: 0x0003 UUID: 6dea7371-0bdf-43db-8ea8-1c3a6a7e5909
GPS Characteristic Value Declaration	0x0003	6dea7371-0bdf-43db-8ea8-1c3a6a7e5909	Read	GPS location data
GPS Descriptor Declaration (CCCD)	0x0004	0x2902	Read & write	Indication = 0x0
MOB Characteristic Declaration	0x0005	0x2803	Read	Properties: Read or indicate Handle of value: 0x0006 UUID: 6dea7372-0bdf-43db-8ea8-1c3a6a7e5909
MOB Characteristic Value Declaration	0x0006	6dea7372-0bdf-43db-8ea8-1c3a6a7e5909	Read	Whether or not the unit has detected that it is submerged in water
MOB Descriptor Declaration (CCCD)	0x0007	0x2902	Read & write	Indication = 0x0

Table 3.9: GSS Attribute Table

The chosen Generic Attribute Profile (GATT) attributes and characteristics are defined in code in Figure 3.25 using macros provided by the Zephyr BLE API.

```c
BT_GATT_SERVICE_DEFINE(gss_svc,
BT_GATT_PRIMARY_SERVICE(BT_UUID_GSS),
           /* GPS Coordinate Characteristic */
           BT_GATT_CHARACTERISTIC(BT_UUID_GSS_GPS,
                        BT_GATT_CHRC_READ,
            BT_GATT_PERM_READ,
                        read_gps_data,
                        NULL,
                        &gps_data),
                /* MOB Event Characteristic */
                BT_GATT_CHARACTERISTIC(BT_UUID_GSS_MOB,
                        BT_GATT_CHRC_READ,
            BT_GATT_PERM_READ,
            read_mob_event_status,
            NULL,
                        &mob_status),
);
```

Figure 3.25: Code to set up GSS

BLE Code in main.c

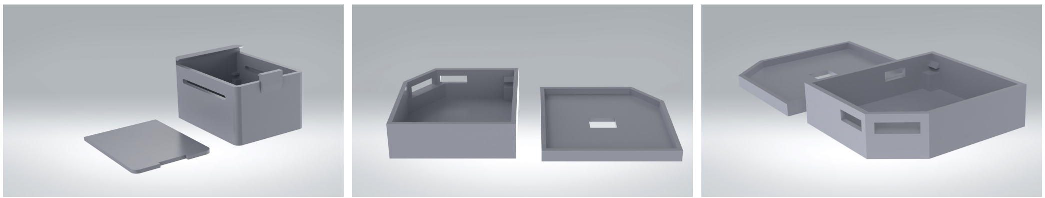
Figure 3.26 below illustrates important sections of the Bluetooth code. The connection callbacks and application callbacks are defined here, bridging the gap between the main.c, gss, and BLE modules, and promoting a modular design.

```c
struct bt_conn_cb connection_callbacks = {
    .connected = on_connected,
    .disconnected = on_disconnected,
};

static gss_cb_s app_callbacks = {
    .gps_cb = getGPSData,
    .mob_cb = getMOBStatus,
};
```

Figure 3.26: BLE code in the main.c

3.3 Encapsulation



(a) Encapsulation for the AUX unit (b) Encapsulation for the Main Unit (c) Encapsulation for the Main Unit

Figure 3.27: Encapsulation designs

Figure 3.27a and Figure 3.27b,3.27c illustrate the encapsulations of the Auxiliary and Main Units respectively. The 3D models of these units were created using SketchUp, a process that involved several design iterations to ensure optimal functionality. Once the final designs were approved, STL files were generated from the SketchUp models.

These STL files were then imported into FlashPrint 5, a slicing software specific to our 3D printer, the Flashforge Finder. The software was used to convert the 3D models into a series of thin layers, preparing them for the 3D printing process.

The 3D printer was then used to construct the cases. This process was monitored closely to ensure the quality and accuracy of the print.

A key aspect of the design was ensuring accurate sensor readings. To this end, ventilation and openings were incorporated into the cases. These served two main functions: they allowed for the smooth operation of the unit by providing accurate measurements from the surroundings, and they permitted water ingress in case of MOB events. This design feature allows the unit's humidity sensor to detect such incidents.

3.4 Soldering

The PCBs were designed to be soldered using a manual pick-and-place technique. Pick-and-place soldering using a stencil is a commonly employed method in the assembly of PCAs. This approach emphasizes precision and requires careful attention to detail. The following are the steps involved in this process:

1. **Stencil Design and Fabrication:** The process starts with the design and creation of a stencil, which is typically a thin sheet of metal or plastic with cut-out patterns that correspond to the solder pads on the PCB. The stencil design is usually derived from the PCB's design files.

2. **Applying solder paste:** The stencil is placed over the PCB, and solder paste — a mixture of tiny solder particles and flux — is spread over it. The solder paste is applied through the cut-outs onto the PCB, depositing solder only where it is needed. This process is often referred to as stencil printing.

3. **Manual Pick-and-Place Process:** Rather than using an automated machine, components are manually picked up using a tweezer and carefully placed onto the corresponding solder-paste-covered pads on the PCB. This step requires a steady hand and meticulous attention to ensure the accurate positioning of each component.

4. **Reflow Soldering:** After all components have been placed accurately, the PCB is moved to a reflow oven. Here, the board is heated to melt the solder paste, thus creating an electrical and mechanical bond between the components and the PCB.

5. **Inspection and Testing:** Once the board has cooled and the solder has solidified, the board undergoes inspection. For smaller-scale production, inspection involves manually checking the PCAs via microscopes. For larger-scale production, this step may involve visual inspection or the use of AOI machines to ensure all components are correctly placed and soldered. Further functional and in-circuit testing may also be conducted to verify the performance of the PCAs.

While the manual pick-and-place soldering using a stencil method requires a high level of precision and care, it offers flexibility and can be more cost-effective for small-scale or custom PCB production. Knowing that the group had to place each component by hand played a role in choosing components.

3.5 Testing

Implementing a robust and detailed testing strategy while examining the various components of a new design can reveal any hidden flaws. While crafting a test plan might initially seem labor-intensive, it can actually streamline the testing phase considerably. A well-structured testing procedure can methodically pinpoint errors and facilitate seamless documentation throughout the process. This stands in contrast to a more casual, ”go-with-the-flow” approach to testing, which often provides less clarity about the design’s strengths and weaknesses. The tests should be comprehensively outlined, providing clear instructions on their execution and establishing pass-criteria in advance. Additionally, documenting results should be an ongoing process during the test execution.

The test plans for the electrical units in are based upon a bottom-up approach with increasing complexity, ensuring that the prerequisites for each test are fulfilled before the test is conducted. The individual test plans for the Main Unit and Auxiliary Unit are given in a shorthand form in Table 3.10 and 3.11

	Test	**Pass Criteria**
T1	Visual inspection	No short-circuits visible through microscope inspection
T2	Continuity test	Electrical continuity test shows no short-circuits
T3	3V3	Correct supply voltage from battery
T4	Power Switch	Battery is able to be disconnected
T5	MCU Alive	MCU boots, is able to receive software
T6	UI functionality	Buttons, Buzzer, and LEDs function as intended
T7	Bluetooth Antenna	Unit is able to connect via Bluetooth communication
T8	Battery Life	Battery meets or exceeds projected life expectancy
T9	Heat Dissipation	System temperature is within the safe range
T10	Signal Strength	Strong and stable signal at minimum distance
T11	Durability Test	Withstands environmental stresses without functional impairment
T12	Software Stability	No crashes or unexpected behavior under various conditions
T13	Interference	Normal operation in a high-interference environment
T14	Usability	Users successfully complete specific tasks
T15	Sensor Functionality	All sensors perform accurately within specified range and conditions

Table 3.10: Testplan Auxiliary Unit

3.5.1 Test Execution

The tests mentioned in the previous section have not been conducted at the time of delivery of this report. This is due to system design, production, and assembly delays.

	Test	Pass Criteria
T1	Visual inspection	No short-circuits visible through microscope inspection
T2	Continuity test	Electrical continuity test shows no short-circuits
T3	3V3 Regulator	Stable 3.3V with <5mV pp ripple
T4	1V8 Regulator	Stable 1.8V with <5mV pp ripple
T5	USB connector	USB type-C connector is able to power up the Unit
T6	Battery charger	Battery is able to be charged via The USB type-C connector
T7	MCU Alive	MCU boots, is able to receive software
T8	UI functionality	Buttons, Buzzer, and LEDs function as intended
T9	Bluetooth Antenna	Unit is able to connect via Bluetooth communication
T10	LTE antenna	Unit is able to connect via LTE communication
T11	GPS antenna	Unit is able to connect via GPS communication
T12	Battery Discharge Rate	Battery discharge rate aligns with projected rate
T13	LTE Signal Strength	Stable connection and data rate at minimum signal strength
T14	GPS Accuracy	Accurate position determination within specified margin
T15	Environmental Resistance	Unit operates normally under various environmental conditions
T16	Software Stability	No crashes or unexpected behavior under various conditions
T17	Interference	Normal operation in a high-interference environment
T18	Usability	Users successfully complete specific tasks
T19	Sensor Functionality	All sensors perform accurately within specified range and conditions

Table 3.11: Testplan Main Unit

Chapter 4

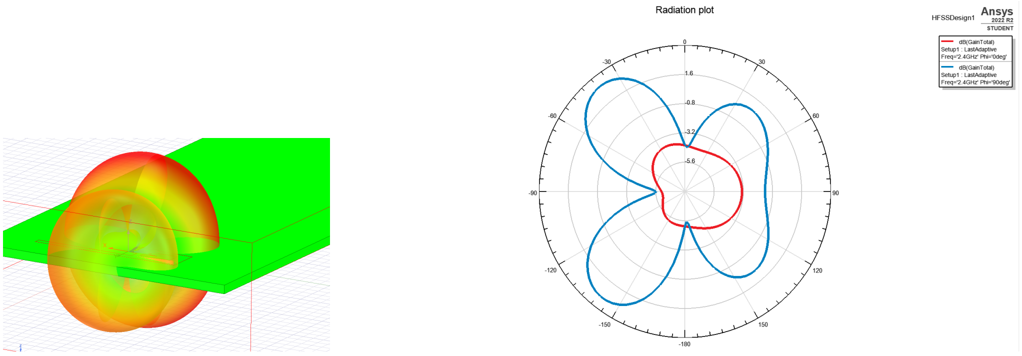
Results

Since we didn't manage to produce and test our prototypes, all firmware was built and tested on the development kits, presented in Chapter 1.3. The schematic of the final design we sent for production can be found in Appendix C.

4.1 Simulation results

4.1.1 Ansys Electrical Simulations

The simulated antenna is shown in Figure 4.1b. Figure 4.2 shows the simulated results. The simulated antenna has minimum return loss of 48.55 dB at 2.395 GHz according to the VSWR graph. Figure 4.1 shows the radiation pattern of the antenna modeled in Ansys Electrical.



(a) Radiation pattern and antenna

(b) Radiation plot

Figure 4.1: Antenna simulated in Ansys Electrical

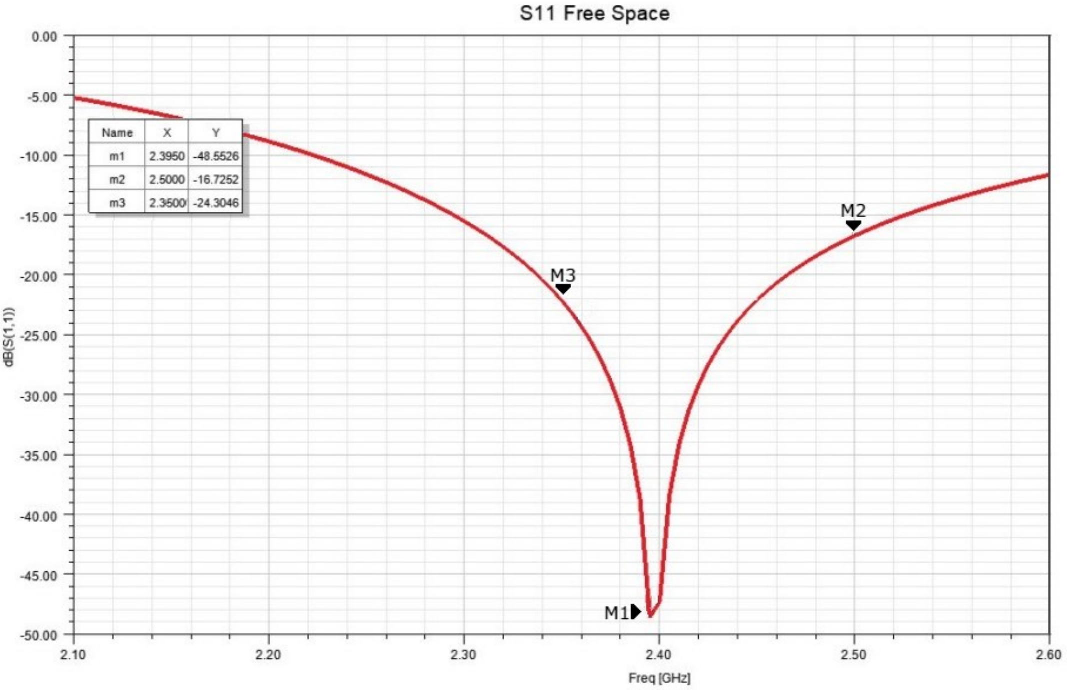


Figure 4.2: Voltage standing Wave Ratio in Ansys

The nRF52832 radio range is marked in Figure 4.3 with markers M1 and M3. The markers show that the frequencies within the radio range are relatively close to the center of the Smith Chart, marked as M2. This suggests that the impedance of the modified antenna is well-matched to the 50 Ω impedance within the radio range, leading to a low amount of reflected power and high efficiency.

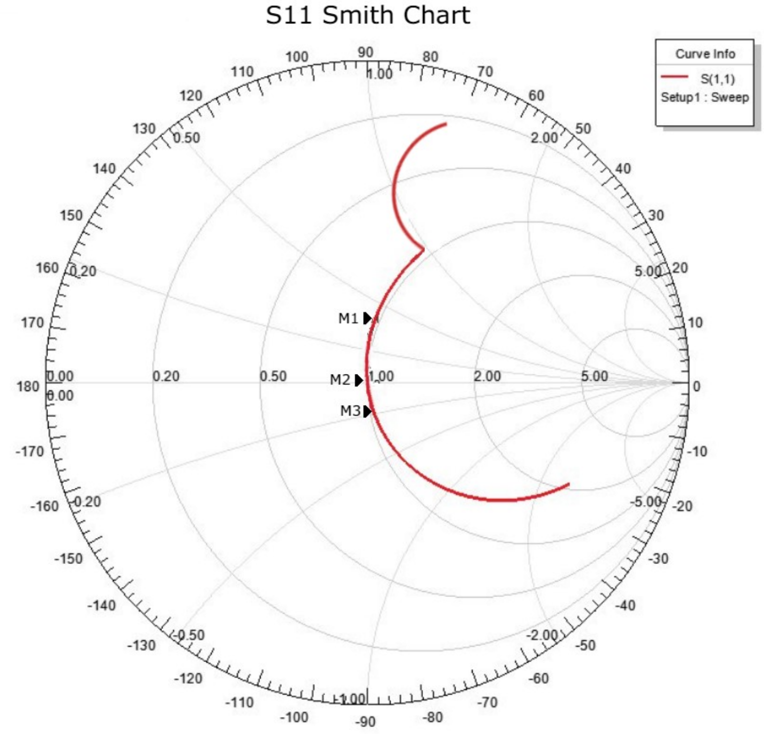


Figure 4.3: Smitch Chart from Ansys Electrical

4.1.2 LTPowerCAD Simulations

The LTM4622 regulator was simulated in LTpowerCAD. These show satisfactory results, with an average efficiency of 92.5% [90]. Figure 4.4 shows the efficiency of the simulated setup. This is discussed further in Chapter 5.3

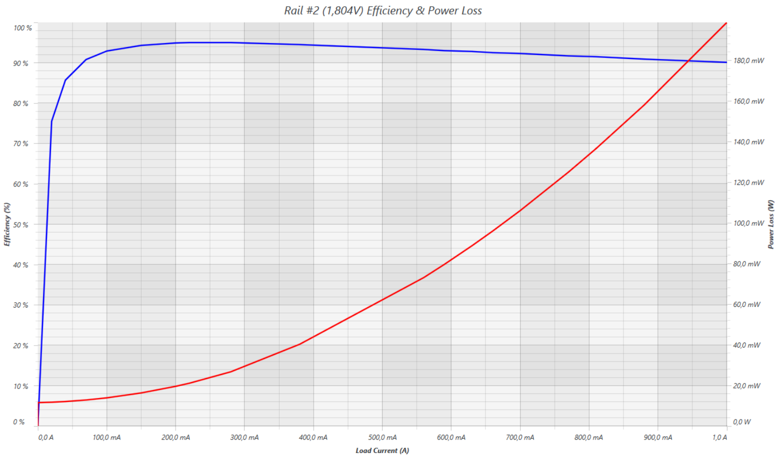
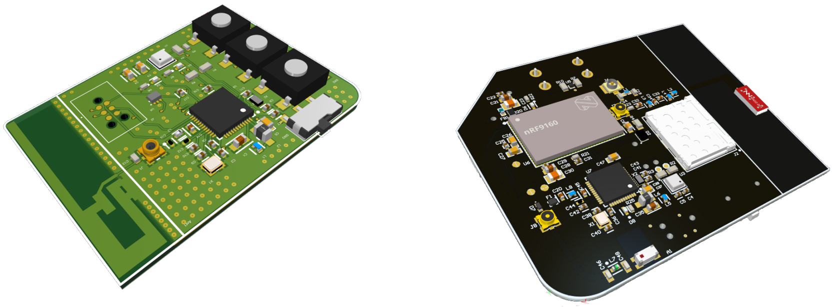


Figure 4.4: LTPowerCAD Efficiency

4.2 Hardware results

The finished prototypes have yet to be produced due to a longer-than-anticipated delivery time, and an unfortunately timed long weekend at Nordic Semiconductor.

Figure 4.5 shows how the prototypes would look like if they were finished. Appendix D shows these in more detail.



(a) Auxiliary Unit

(b) Main Unit

Figure 4.5: PCB

4.3 Firmware Results

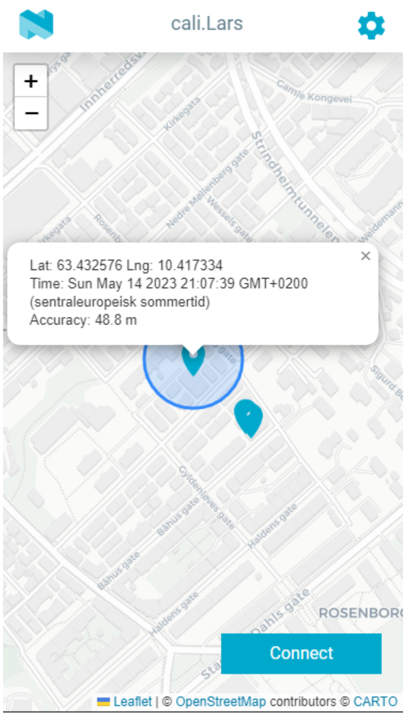
We were unfortunately not able to implement our firmware on our PCBs, so the software we developed have been tested on the nRF9160 DK and nRF52840 DK. This section presents the firmware results, detailing the internal communication, sensor readings, GNSS, and LTE-M implementation.

4.3.1 GNSS

GNSS/GPS was implemented on the Main Unit and Vessel Unit, as these had GNSS capabilities. The code running on these are similar in multiple places, resulting in a similar log output.

Figure 4.7 presents the terminal output from the Main Unit, which begins at startup and concludes with the acquisition of the first two GNSS fixes and the subsequent transmission of this data to a CoAP server. The output illustrates the operations of the system in real-time. The events displayed confirm the successful initialization and activation of the GNSS, the transition of the device status to 'searching' and then to 'fixed', and the successful acquisition of GNSS fix data.

Figure 4.6 visualizes an acquired position data on a map. This geographic representation confirms the successful translation of the raw GNSS data into usable location information.



Lat: 63.432576 Lng: 10.417334
Time: Sun May 14 2023 21:07:39 GMT+0200
(sentraleuropeisk sommertid)
Accuracy: 48.8 m

Figure 4.6: Screenshot of position visualized on a map

These results demonstrate the successful operation of the GNSS tracker, from the initial system startup to the acquisition of the first GNSS fix.

4.3.2 LTE-M Implementation

The implementation of the LTE-M was successful. The implementation ensured that the system switched between LTE and GNSS.

Functionality and Synchronization

The `lte_handler()` function managed LTE events, notably checking the network registration status. If the device was registered to a network, it logged the status and released the `lte_connected` semaphore. The `modem_configure()` function initialized and connected to the LTE network and deactivated LTE to enable GNSS. If any step failed, it logged an error message.

LTE Activation and Deactivation

The `main()` function in the main.c file attempted to activate LTE using `lte_lc_func_mode_set(LTE_LC_FUNC_MODE_NORMAL)`. If this operation failed, an error message was logged and the loop broke. After handling CoAP communication, the program tried to deactivate LTE and enable the GNSS functional mode using `lte_lc_func_mode_set(LTE_LC_FUNC_MODE_DEACTIVATE_LTE)`. If this operation failed, an error message was logged and the loop broke. If an error caused the program to break out of the loop, `lte_lc_func_mode_set(LTE_LC_FUNC_MODE_NORMAL)` was called to attempt to reactivate LTE before the system loop started again.

LTE-GNSS Integration and Data Transmission

The integration of the LTE-M and GNSS allowed for efficient tracking and data transmission. The GNSS, when fix was achieved, sent the location data to the LTE-M for transmission to the server. The program handled these interactions seamlessly, ensuring that the two systems did not interfere with each other and maximizing efficiency. The success of this integration was a significant achievement in the project.

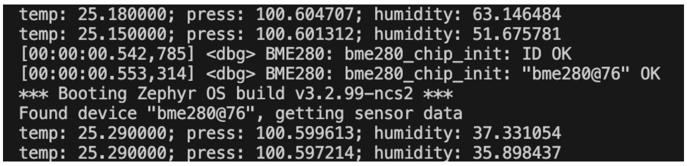
Additionally, the successful resolution of the server address and subsequent data transmission over LTE demonstrates performance of the communication protocols in place.

```
[00:00:00.569,000] <inf> main_c_: Main Unit Version 1.0.0 started
[00:00:00.569,000] <inf> led_button: Initializing LEDs and buttons
[00:00:00.569,091] <inf> main_c_: Initializing LTE for upload....
[00:00:00.902,435] <inf> LTE: LTE cell changed: Cell ID: -1, Tracking area: -1
[00:00:01.046,600] <inf> main_c_: Starting GNSS....
[00:00:01.073,394] <inf> main_c_: GNSS started successfully
[00:00:01.073,516] <dbg> sensors: sensors_init: BME280 device initialized.
[00:00:01.073,547] <dbg> sensors: sensors_init: No ADXL362 device found.
[00:00:01.073,577] <dbg> sensors: sensors_init: No ADXL372 device found.
[00:00:01.073,577] <wrn> sensors: Some sensors failed to initialize.
...
[00:00:57.471,160] <dbg> GNSS: gnss_event_handler: Searching for GNSS Satellites....
[00:00:58.492,492] <inf> GNSS: GNSS fix event
[00:00:58.493,804] <inf> GNSS: GNSS enters sleep because fix was achieved in periodic mode
                               Latitude:       63.431872
                               Longitude:      10.419127
                               Altitude:       102.0 m
                               Time (UTC):     17:20:30.387
[00:00:58.494,049] <inf> main_c_: Sending data over LTE
[00:00:58.530,303] <inf> main_c_: Waiting for LTE connection
[00:00:59.300,750] <dbg> main_c_: sensor_thread: BME280: Temperature: 18.490000 C,
                                  Pressure: 102.397050 hPa, Humidity: 52.891601 %RH
[00:00:59.531,036] <inf> LTE: LTE cell changed: Cell ID: 21673474, Tracking area: 30401
[00:00:59.584,075] <inf> LTE: RRC mode: Connected
[00:01:01.135,253] <inf> LTE: Network registration status: Connected - roaming
[00:01:01.135,559] <inf> main_c_: LTE connected
[00:01:01.135,559] <inf> main_c_: Resolving the server address
[00:01:01.136,657] <inf> LTE: PSM parameter update: TAU: 1152000, Active time: 20
[00:01:01.260,345] <inf> CoAP: IPv4 Address found 20.47.97.44
[00:01:01.260,345] <inf> main_c_: Sending Data over LTE
...
[00:01:09.344,879] <dbg> GNSS: gnss_event_handler: Searching for GNSS Satellites....
[00:01:09.345,947] <inf> GNSS: GNSS enters sleep because fix was achieved in periodic mode
                               Latitude:       63.431993
                               Longitude:      10.418474
                               Altitude:       69.5 m
                               Time (UTC):     17:20:41.242
[00:01:09.346,191] <inf> main_c_: Sending data over LTE
[00:01:09.382,446] <inf> main_c_: Waiting for LTE connection
...
[00:01:12.071,533] <inf> CoAP: CoAP request sent: token 0x9999
[00:01:12.151,275] <inf> CoAP: CoAP response: Code 0x44, Token 0x9999,
                               Payload: 63.431993,10.418474
27.5 m
2023-05-21 17:20:41
0
```

Figure 4.7: Terminal output showing GNSS data and connection to a CoAP server through LTE-M

4.3.3 Serial Communication

Figure 4.8 shows the terminal output with sensor readings from the BME280 sensor.

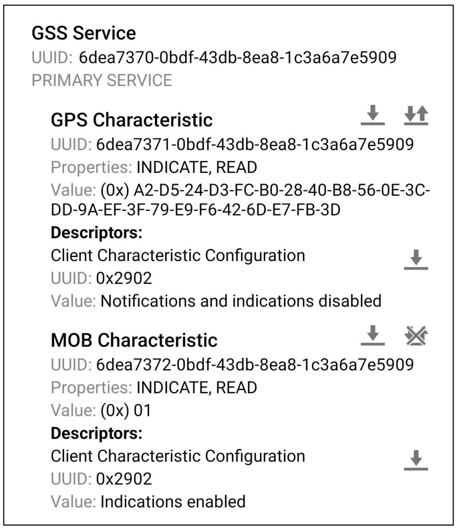


```
temp: 25.180000; press: 100.604707; humidity: 63.146484
temp: 25.150000; press: 100.601312; humidity: 51.675781
[00:00:00.542,785] <dbg> BME280: bme280_chip_init: ID OK
[00:00:00.553,314] <dbg> BME280: bme280_chip_init: "bme280@76" OK
*** Booting Zephyr OS build v3.2.99-ncs2 ***
Found device "bme280@76", getting sensor data
temp: 25.290000; press: 100.599613; humidity: 37.331054
temp: 25.290000; press: 100.597214; humidity: 35.898437
```

Figure 4.8: Terminal during testing phase

4.3.4 Internal Communication

Figure 4.9 illustrates the service with two characteristics on nRF Connect for Mobile. This functionality was key in enabling efficient communication between different elements of the tracking system.



GSS Service
UUID: 6dea7370-0bdf-43db-8ea8-1c3a6a7e5909
PRIMARY SERVICE

GPS Characteristic
UUID: 6dea7371-0bdf-43db-8ea8-1c3a6a7e5909
Properties: INDICATE, READ
Value: (0x) A2-D5-24-D3-FC-B0-28-40-B8-56-0E-3C-DD-9A-EF-3F-79-E9-F6-42-6D-E7-FB-3D
Descriptors:
Client Characteristic Configuration
UUID: 0x2902
Value: Notifications and indications disabled

MOB Characteristic
UUID: 6dea7372-0bdf-43db-8ea8-1c3a6a7e5909
Properties: INDICATE, READ
Value: (0x) 01
Descriptors:
Client Characteristic Configuration
UUID: 0x2902
Value: Indications enabled

Figure 4.9: GSS Service on a receiver

Figure 4.10 shows the terminal output from the Main Unit or Auxiliary Unit when a receiver subscribes and unsubscribes to the characteristics MOB event and GPS tracking.

```
[00:03:02.783,264] <inf> conn: Connected
[00:03:07.286,621] <inf> gss: GPS indication enabled
[00:03:09.702,850] <inf> gss: MOB event indication enabled
[00:03:15.260,284] <inf> gss: MOB event indication disabled
[00:03:15.942,779] <inf> gss: GPS indication disabled
```

Figure 4.10: Terminal output when a receiver subscribes and unsubscribes

4.3.5 Distress Signal

The group were able to implement a distress signal in the form of a boolean variable that turns true when the measured humidity is larger than a certain threshold. This humidity level is measured by a BME280 over an I2C interface, as we did were not able to perform on our PCBs as planned. This boolean variable, called a mob_event internally in the program, is sent along with the latest location data to the CoAP server. As soon as the threshold is reached, it immediately sends out a CoAP message. This signal is reset by pressing `button2`. The serial output from the Main Unit as the humidity threshold is reached can be seen below in Figure 4.11. The ’1’ at the end of the CoAP message payload is the boolean distress signal. This signal can be picked up by other units listening to the same CoAP server.

```
[00:03:46.036,071] <dbg> main_c_: sensor_thread: BME280: Temperature: 18.330000 C,
                                  Pressure: 102.395175 hPa, Humidity: 61.406250 %RH
[00:03:46.036,071] <wrn> main_c_: MOB ALERT!
[00:03:46.036,132] <inf> main_c_: Sending data over LTE
[00:03:46.074,554] <inf> main_c_: Waiting for LTE connection
...
[00:03:48.270,355] <inf> main_c_: LTE connected
[00:03:48.270,385] <inf> main_c_: Sending Data over LTE
[00:03:48.637,481] <inf> CoAP: CoAP request sent: token 0x2520
[00:03:48.720,397] <inf> CoAP: CoAP response: Code 0x44, Token 0x2520, Payload:
63.431901,10.418642
25.5 m
2023-05-21 17:22:40
1   <-- Distress signal
[00:03:48.900,573] <inf> LTE: LTE cell changed: Cell ID: -1, Tracking area: -1
...
```

Figure 4.11: Distress signal

Chapter 5

Discussion

This thesis prioritized power optimization within the Main Unit, guiding the hardware design and firmware development processes.

Due to unexpected delays in the arrival of the manufactured prototypes, a change in approach became necessary. This unforeseen challenge provided an opportunity to adjust and refine the prototypes via simulations and further use of the development kits.

The decision to primarily focus on conducting detailed simulations of the prototype designs offered several benefits. It provided the ability to gain valuable insights, test functionality, and troubleshoot potential issues, all within a controlled, virtual environment. Despite not being physical models, these thorough simulation processes yielded essential data, which enhanced the performance and efficiency of the prototypes.

During the development phase, we ended up utilizing the nRF9160DK and nRF52840DK combined as the Vessel Unit, instead of a Thingy:91 as we originally planned. This was because of mainly two factors. First, the Thingy:91 we used for testing gave some error codes that slowed down the process of working with it. Second, we didn't flesh out the BLE code for the Vessel Unit as much as we planned to, which was the main purpose of using the Thingy:91 as this unit.

5.1 Simulations

5.1.1 Ansys Electrical Simulations

A simulated PIFA antenna was analyzed in Ansys Electrical. Figure 4.2 suggests that the antenna is transmitting the majority of the incident power at that frequency. The simulations showed the efficient reflection is 24.30 dB at 2.35 GHz and around 16.73 dB at 2.50 GHz. After calculating these values, the peak efficiency ended up being 97.4% and the minimum efficiency was 95.8%. These calculations were done within the nRF52832 radio range of 2.35 GHz to 2.50 GHz [86].

Such a high efficiency implies that the majority of the input power is being radiated by the antenna. With a peak efficiency of 97.4 % and a minimum efficiency of 95.8% within the frequency range of the nRF52832 radio. This implies that the antenna could transmit a high portion of the input power at all frequencies within the radio range.

The simulation results provided information that could be used to tune the actual antenna. For example, the team was able to determine the optimal dimensions of the antenna and the frequency range in which it performed best. These insights were crucial in optimizing the performance of the antenna.

However, it is important to note that there is no guarantee that the changes made to the parameters in the simulation would result in the same performance in the real world. The simulation only provides a theoretical prediction of the antenna's behavior, and there may be additional factors in the real world that affect the performance of the antenna. Since the physical units were never tested, such a comparison could not be made.

5.1.2 LTpowerCAD Simulations

The Simulations from LTpowerCAD showed that with the intended battery's nominal voltage at 3.7V, the regulator would perform below average. When using the 5V supplied by the USB type-C connector, most of the performance issues were solved. As this unit is intended to be used mostly with the battery, this issue would have to be resolved for later revisions, but would most likely not be an issue during the development and testing of the prototype.

Upon going through the LTpowerCAD simulations of the LTM4622, it was discovered that the circuit could have a tendency to become unstable. In an ideal scenario, the gain margin is expected to be below -8 dB and the phase margin should exceed 45° [91]. As seen in Figure 5.1, the phase margin was in the clear at 50.2°, but the gain margin was too high at -28.4 dB. This could cause the power rail to exhibit high output voltage ripple, as illustrated in the Texas Instruments Design Seminar [92].

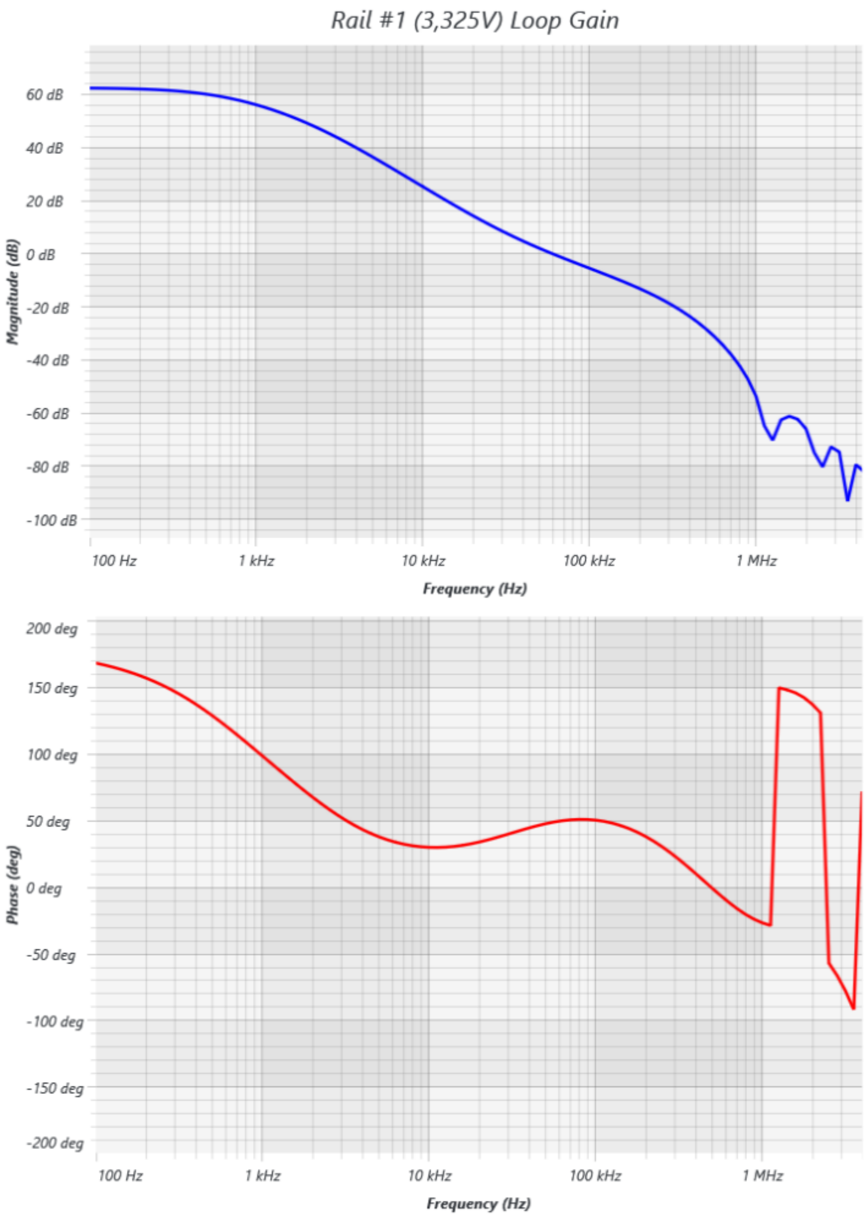


Figure 5.1: Gain and Phase plots from LTpowerCAD

Another potential issue could be the inductor ripple current when V_{IN} was below 4.1V. The Summary for the LTpowerCAD can be found in the Appendix E.

5.2 Firmware

One of the biggest challenges while working on the firmware was the relatively small developer community and online resources. We used the Nordic DevZone forums [93] frequently to ask questions, as well as asking our contact person directly.

Our modular approach based on the different functionalities proved efficient, as the difference between the Main and Auxiliary Units were the cellular communication on the Main Unit. This let us develop firmware in parallel of each other simultaneously.

The LTE-M was setup on the Main Unit successfully, and we managed to send the data we wished to a CoAP server.

The GNSS implementation, as evidenced by the results, was successful in achieving accurate GNSS fixes in outdoor environments. The system successfully initialized and activated the GNSS, transitioned the device status from 'searching' to 'fixed', and successfully captured GNSS fix data. This functionality was achieved through a concurrent operation of the GNSS thread and the main thread, which were synchronized using a semaphore. The GNSS priority mode enabled the system to prioritize the GNSS receiver over the LTE modem, ensuring the successful operation of both.

However, our testing was constrained, and there was a missed opportunity to conduct extensive field trials, particularly in marine environments. Comprehensive testing would have provided a broader understanding of the precision and reliability of our GNSS data. This could have guided potential modifications in our solution, impacting power usage and enhancing the real-world applicability of our device.

Furthermore, while the GNSS-LTE alternating strategy is effective in power optimization, it is essential to recognize that it may not be the optimal solution in all scenarios. The GNSS fix time can vary significantly depending on the environmental conditions, which may lead to longer waiting times and higher power consumption for the main thread. Therefore, more research is needed to explore power-efficient strategies for GNSS operation in different contexts.

The design of the auxiliary unit, conceived and developed as an additional feature, enhances the surveillance capabilities of the vessel system. Although not part of the original specifications, it provides extra value by serving as a potential backup for the Main unit in case of unexpected outages. It acts as a modular extension, engineered to integrate with existing units without compromising their functionality or performance. This unit's design promotes scalability, interchangeability, and easy maintenance, making it a robust, flexible addition to the system.

However, one crucial aspect that has been less than optimal during the design phase of the auxiliary unit is the extent of testing. Comprehensive testing is critical to identify potential faults, verify design assumptions, and validate the reliability and robustness of the unit. In the case of the auxiliary unit, the amount of testing carried out was insufficient, mainly due to constraints on the project timeline and resources.

This lack of extensive testing could potentially expose the system to unidentified faults or performance issues in real-world operations. Hence, it is highly recommended for future work on the project to prioritize comprehensive and rigorous testing of the auxiliary unit. This will ensure that it meets the necessary performance and reliability standards and is fit for integration into the larger system.

Since we didn't get our own hardware, we didn't test the power consumption through the Power Profiler Kit II. We wanted to use it to gather data on our own products running our own firmware, but since it didn't arrive on time this part of the project was omitted.

The encapsulation had some critical faults, rendering it useless in this project. The production of these were therefore scrapped.

5.3 Further work

5.3.1 Main Unit - Hardware

Change regulator

As described under section 5.1.2, the LTM4622 is a sub-optimal regulator for usage at the battery's nominal voltage. It was not done enough research while looking for a regulator. After the PCBs were sent to production, the group concluded that using 2 simpler regulators, such as the LTC3549 [94], could have provided a more stable voltage at a lower nominal voltage. This would come at the cost of more complexity and a larger overall size.

Implement USB

Due to the lack of data lines on the nRF52832 [86], USB communications were never established. A solution was temporarily put in place, with a USB to UART translater chip[95], but was ultimately removed as the group saw no benefit of using the USB. If a different nRF chip is used, for example the Nordic nRF5340 [96], data transfer via USB could be implemented. If not using the USB to UART translator described in Figure 3.5 is also an option.

Testing different functionalities

The test plan describes in table 3.11 should be expanded upon, and performed on the produced units.

Future implementations

There are several areas in which the current man overboard sensor system can be improved and expanded upon to provide better performance and additional features. The following list outlines some potential enhancements and areas of further research:

- **Waterproof Casing:** In order to ensure the longevity and reliability of the system in marine environments, a waterproof casing must be developed. This casing should be designed to protect the sensor and other electronic components from water ingress, salt, and other corrosive elements found in marine environments. Additionally, the casing should be robust enough to withstand rough handling and impact forces that may be experienced during normal operation on a vessel.
- **Smart Power Management:** The current design of the man overboard sensor may benefit from the integration of smart micromodules for more efficient power management. These modules can monitor the power consumption of the device and adjust its operation to optimize battery life. For example, the device could enter a low-power mode when not in use or when the vessel is stationary. This would extend the operational life of the sensor and reduce the frequency of battery replacement or recharging.
- **Improved Detection Algorithms:** As the field of machine learning and artificial intelligence continues to advance, more sophisticated algorithms can be developed to improve the accuracy and reliability of the man overboard detection system. By incorporating new techniques, the sensor could potentially reduce false alarms and improve its ability to identify genuine man overboard events. Implementing this would take a lot of time and resources.
- **Integration with Existing Safety Systems:** Investigating the compatibility of the man overboard sensor with existing maritime safety systems and protocols can help streamline the response process and provide a more comprehensive safety solution. This could involve integrating the sensor with automatic identification systems (AIS), vessel monitoring systems (VMS), or other safety equipment onboard the vessel.

5.3.2 Auxiliary Unit - Hardware

Testing different functionalities

The test plan describes in table 3.10 should be expanded upon, and performed on the produced units.

Enhanced Power Management and Protection

While the initial design has successfully integrated a CR2032 battery as the power source, there are several potential improvements that could be considered for future iterations of the device.

1. **Voltage Regulation:** While the nRF52832 microcontroller operates satisfactorily at the battery's nominal 3.3V, the introduction of a voltage regulator could help to stabilize the voltage supply during battery discharge. This would ensure consistent performance even as the battery nears the end of its charge.

2. **Under-Voltage Lockout (UVLO):** To prevent the battery from deep discharging, an under-voltage lockout circuit could be implemented. This would automatically disconnect the load if the battery voltage drops below a certain threshold, protecting the battery from damage and prolonging its lifespan.
3. **Battery Level Detection:** Future designs could also consider integrating a battery level detection circuit. This would give users a visual indication of the remaining battery life, allowing them to predict when the battery needs to be replaced. This could be achieved by connecting a voltage divider network to one of the microcontroller's ADC channels.
4. **Sleep Mode:** To increase battery life, the device could make use of the nRF52832's power management features to implement a sleep mode. This would switch off non-essential functions and peripherals when they're not in use, greatly reducing power consumption during idle periods.
5. **Resettable Fuse:** To protect the device's sensitive electronics from potential short circuits, a resettable fuse could be added in series with the battery. This would disconnect the circuit in the event of a current spike, and automatically reset once the condition is resolved.

By addressing these areas, future iterations of the device could offer improved performance and reliability, while also extending the life of the battery.

5.3.3 Firmware

Further work on the be done on the firmware is:

- Develop UART for communication between the nRF9160 and nRF52832.
- Set up the Thingy:91 as a complete Vessel Unit.
- Transfer the code and make it work on our PCBs.
- Setup of our own CoAP server.
- Utilize interrupts on the BME680 sensor, such that contact with water immediately triggers a response, without having a thread polling.
- Integrate Bluetooth LE directly with GPS and sensor polling.

Chapter 6

Conclusion

In conclusion, this project has made significant strides in developing a prototype system for the Main, Auxilary and Vessel Units. Both hardware and software aspects were addressed, resulting in a compact design as portrayed in figure 6.1, which showcases the produced PCBs. Furthermore, the key specifications of the design are summarized in table 6.1.

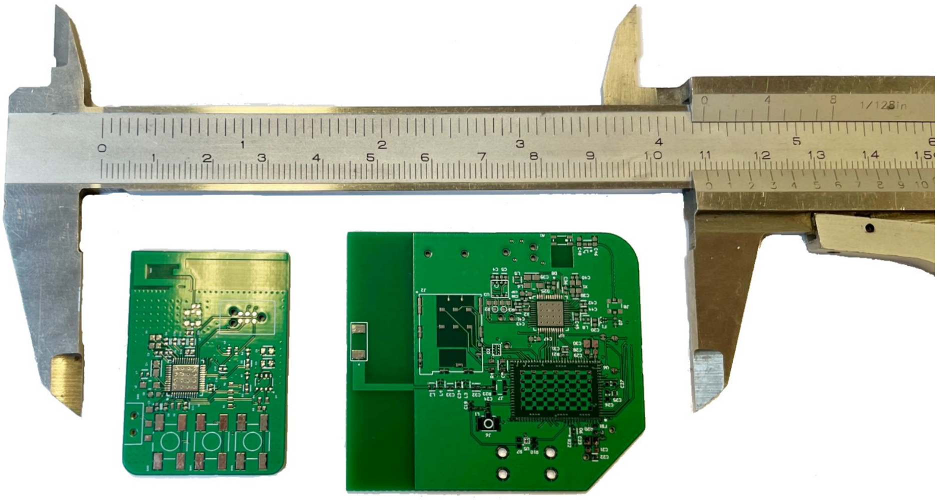


Figure 6.1: Produced PCBs. Caliper for scale

The current hardware design presents a foundation upon which further development can be built.

Despite the promising results, it is important to note the limitations and constraints of the current project. Due to time constraints, a fully functioning prototype could not be developed and tested extensively, which represents the most significant limitation of this project. Future work, as

	Dimension [mm]	Component count
Main Unit	53.9 x 47	127
Auxilary Unit	40.2 x 28.7	45

Table 6.1: Key specifications for both units

outlined in Section 5.3, presents opportunities for further development and refinement of this technology. These areas for improvement not only highlight the potential for enhancement of the current work but also open avenues for further research and exploration

The firmware will have to be revised before transferred to the designed prototypes. Although we managed to set up and test a lot of the functions in isolation, a lot of work is yet to be done to make this deployable in the field.

The journey to the creation of this prototype system was marked by both challenges and victories. However, the experience gained, and the steps taken towards the creation of a working design, reaffirm the potential of this project. The aspiration moving forward is to take the experienced and we have learned from this phase of the project, refine the prototype further, and ultimately deliver a fully functional, reliable, and efficient system.

Bibliography

[1] C. Hine, 'Cellular iot asset tracking applications gain momentum', *Get Connected Blog*, 13th Oct. 2021.

[2] Nordic Semiconductor. 'Nrf9160 dk'. (), [Online]. Available: https://www.nordicsemi.com/Products/Development-hardware/nrf9160-dk (visited on 14th Apr. 2023).

[3] Nordic Semiconductor. 'Nrf52840 dk'. (), [Online]. Available: https://www.nordicsemi.com/Products/Development-hardware/nrf52840-dk (visited on 14th Apr. 2023).

[4] Nordic Semiconductor. 'Nordic thingy:91'. (24th Sep. 2022), [Online]. Available: https://www.nordicsemi.com/Products/Development-hardware/Nordic-Thingy-91 (visited on 4th Apr. 2023).

[5] Nordic Semiconductor. 'Power profiler kit ii'. (), [Online]. Available: https://www.nordicsemi.com/Products/Development-hardware/Power-Profiler-Kit-2 (visited on 14th Apr. 2023).

[6] Nordic Semiconductor. 'Nordic developer academy'. (), [Online]. Available: https://www.nordicsemi.com/academy (visited on 15th Mar. 2023).

[7] Nordic Semiconductor. 'Nrf connect sdk fundamentals'. (), [Online]. Available: https://academy.nordicsemi.com/courses/cellular-iot-fundamentals/ (visited on 21st Mar. 2023).

[8] Nordic Semiconductor. 'Bluetooth low energy fundamentals'. (2022), [Online]. Available: https://academy.nordicsemi.com/courses/bluetooth-low-energy-fundamentals/ (visited on 3rd May 2023).

[9] Nordic Semiconductor. 'Cellular iot fundamentals'. (2022), [Online]. Available: https://academy.nordicsemi.com/courses/cellular-iot-fundamentals/ (visited on 4th Apr. 2023).

[10] M. Revheim-Rafaelsen and K. Rørslett. 'Vil ha nullvisjon for drukning: Var selv nær ved å dø på havet'. (18th Jan. 2023), [Online]. Available: https://www.nrk.no/norge/vil-ha-nullvisjon-for-drukning_-var-selv-naer-ved-a-do-pa-havet-1.16260725 (visited on 14th Feb. 2023).

[11] T. Grønning. 'Sjømannsforbundet vil at alle skal ha samme sikkerhetsutstyr som kai ronny'. (6th Feb. 2023), [Online]. Available: https://www.nrk.no/tromsogfinnmark/vil-ha-ais-sendere-pabudt-etter-regjeringens-nullvisjon-1.16281856 (visited on 14th Feb. 2023).

[12] Nordic Semiconductor. 'Low power cellular iot'. (), [Online]. Available: https://www.nordicsemi.com/Products/Low-power-cellular-IoT/Optimized-for-low-power?lang=en#infotabs (visited on 18th Mar. 2023).

[13] M. Yogendrappa, 'Why fr4 material is commonly used in pcb fabrication?', *Sierra Circuits*, 27th Jul. 2021.

[14] RayPCB. 'How to plan multilayer pcb stackup'. (8th Mar. 2023), [Online]. Available: https://www.raypcb.com/how-to-plan-multilayer-pcb-stackup/ (visited on 9th Mar. 2023).

[15] Fibre Glast. 'What are prepregs?' (8th Mar. 2023), [Online]. Available: https://www.fibreglast.com/product/about-prepregs (visited on 9th Mar. 2023).

[16] Sierra circuits. 'What is a pcb transmission line?' (7th Mar. 2023), [Online]. Available: https://www.protoexpress.com/blog/pcb-transmission-line/ (visited on 7th Mar. 2023).

[17] Z-Zero. 'Understanding key differences between the time and frequency domains.' (7th Apr. 2023), [Online]. Available: https://www.z-zero.com/blog-post/an-ode-to-pi/ (visited on 7th Apr. 2023).

[18] ANALOG DEVICES. 'What is vswr'. (2023), [Online]. Available: https://www.analog.com/en/design-center/glossary/vswr.html#.ZF43Lh9Opm0.link (visited on 4th May 2023).

[19] Wikipedia. 'Smith chart'. (8th Mar. 2023), [Online]. Available: https://en.wikipedia.org/wiki/Smith_chart (visited on 3rd Mar. 2023).

[20] A. S. Sedra and K. C. Smith, *Microelectronic Circuits.* Oxford University Press.

[21] Antenna Tutorial. 'Maxwell's equation'. (), [Online]. Available: https://www.antenna-theory.com/definitions/maxwellsequations.php (visited on 6th Apr. 2023).

[22] NASA. 'Anatomy of an electromagnetic wave'. (), [Online]. Available: https://science.nasa.gov/ems/02_anatomy (visited on 9th Mar. 2023).

[23] J. D. Kraus and R. J. Marhefka, *Antennas: For All Applications*, 3rd ed. McGraw-Hill, 2002, ISBN: 978-0-07-232103-6.

[24] S. Haykin, *Communication Systems*, 5th ed. John Wiley & Sons, 2009, ISBN: 978-0-470-25388-7.

[25] Tutorialspoint. 'Antenna theory - isotropic radiation'. (), [Online]. Available: https://www.tutorialspoint.com/antenna_theory/antenna_theory_isotropic_radiation.htm (visited on 15th May 2023).

[26] Antenna-Theory. 'The dipole antenna'. (), [Online]. Available: https://www.antenna-theory.com/antennas/dipole.php (visited on 15th May 2023).

[27] C. A. Balanis, *Antenna Theory: Analysis and Design*, 4th ed. John Wiley & Sons, 2016, ISBN: 978-1-118-64279-6.

[28] W. L. Stutzman and G. A. Thiele, *Antenna Theory and Design*, 3rd ed. John Wiley & Sons, 2012, ISBN: 978-0-470-57664-9.

[29] Texas Instruments. 'Application report 2.4-ghz inverted f antenna'. (2022), [Online]. Available: https://www.ti.com/lit/an/swru120d/swru120d.pdf (visited on 3rd Mar. 2023).

[30] Calculator Academy Team. 'Ripple voltage calculator'. (7th Mar. 2023), [Online]. Available: https://calculator.academy/ripple-voltage-calculator/ (visited on 7th Mar. 2023).

[31] Science Direct. 'Ripple factor'. (7th Mar. 2023), [Online]. Available: https://www.sciencedirect.com/topics/engineering/ripple-factor (visited on 7th Mar. 2023).

[32] M. Brown, *Power Supply Cookbook.* Newnes, 2001.

[33] Skillbank. 'John errington's tutorial on power supply design'. (), [Online]. Available: https://skillbank.co.uk/psu/ripple.htm (visited on 7th May 2023).

[34] PCB buy. 'Manhattan routing in pcb manufacturing'. (12th Jan. 2021), [Online]. Available: https://www.pcbbuy.com/news/What-is-Manhattan-Routing-in-PCB-Manufacturing-Process.html (visited on 19th May 2023).

[35] Cadence. 'The challenges of right-angle pcb traces'. (7th Mar. 2023), [Online]. Available: https://resources.pcb.cadence.com/blog/2022-the-challenges-of-right-angle-pcb-traces (visited on 7th Mar. 2023).

[36] Nordic Semiconductor. 'Nrf9160'. (24th Sep. 2022), [Online]. Available: https://www.nordicsemi.com/products/nrf9160 (visited on 6th Apr. 2023).

[37] Nordic Semiconductor. 'Nrf9160'. (4th Oct. 2022), [Online]. Available: https://infocenter.nordicsemi.com/topic/ug_thingy91/UG/thingy91/hw_description/hw_antenna_tuning.html (visited on 4th Apr. 2023).

[38] Nordic Semiconductor. 'Antenna performance'. (4th Oct. 2022), [Online]. Available: https://infocenter.nordicsemi.com/topic/ug_thingy91/UG/thingy91/hw_description/hw_antenna_performance.html (visited on 4th Apr. 2023).

[39] Nordic Semiconductor, 'Nrf9160 product specification v 2.1', 14th Oct. 2021.

[40] Nordic Semiconductor. 'What is cellular iot?' (), [Online]. Available: https://shorturl.at/lrwzG (visited on 30th Apr. 2023).

[41] J. Ávila Rodríguez, 'Gps signal plan', *University FAF Munich, Germany*, 2011, Advanced.

[42] J. Ávila Rodríguez, 'Qzss signal plan', *University FAF Munich, Germany*, 2011, Advanced.

[43] Nordic Semiconductor. 'Global navigation satellite system (gnss)'. (), [Online]. Available: https://academy.nordicsemi.com/topic/lesson-6-gnss/ (visited on 13th Apr. 2023).

[44] Nordic Semiconductor. 'Gnss interface - operation mode'. (19th May 2023), [Online]. Available: https://developer.nordicsemi.com/nRF_Connect_SDK/doc/latest/nrfxlib/nrf_modem/doc/gnss_interface.html#operation-mode (visited on 26th Mar. 2023).

[45] C. Coleman. 'A practical guide to ble throughput'. (24th Sep. 2019), [Online]. Available: https://interrupt.memfault.com/blog/ble-throughput-primer (visited on 4th Apr. 2023).

[46] Nordic Semiconductors. 'What is bluetooth le?' (2023), [Online]. Available: https://academy.nordicsemi.com/topic/what-is-bluetooth-le/ (visited on 6th Apr. 2023).

[47] Nordic Semiconductors. 'Gap: Device roles and topologies'. (2023), [Online]. Available: https://academy.nordicsemi.com/topic/gap-device-roles-and-topologies/ (visited on 6th Apr. 2023).

[48] Nordic Semiconductors. 'Phy: Radio modes'. (2023), [Online]. Available: https://academy.nordicsemi.com/topic/phy-radio-modes/ (visited on 8th Apr. 2023).

[49] Nordic Semiconductors. 'Bluetooth le advertising'. (2023), [Online]. Available: https://academy.nordicsemi.com/lessons/lesson-2-bluetooth-le-advertising/ (visited on 8th Apr. 2023).

[50] Nordic Semiconductors. 'Security in bluetooth le communication'. (2023), [Online]. Available: https://academy.nordicsemi.com/lessons/lesson-5-bluetooth-le-security-fundamentals/ (visited on 8th Apr. 2023).

[51] J. Valdez and J. Becker, 'Understanding the i2c bus', Texas Instruments, Dallas, Texas, Tech. Rep., Jun. 2015.

[52] Nordic Semiconductor. 'I2c protocol'. (), [Online]. Available: https://academy.nordicsemi.com/topic/i2c-protocol/ (visited on 25th Apr. 2023).

[53] SparkFun Electronics. 'Serial peripheral interface (spi)'. (), [Online]. Available: https://learn.sparkfun.com/tutorials/serial-peripheral-interface-spi/all (visited on 10th May 2023).

[54] E. Peña and M. G. Legasi. 'Uart: A hardware communication protocol understanding universal asynchronous receiver/transmitter'. (), [Online]. Available: https://www.analog.com/en/analog-dialogue/articles/uart-a-hardware-communication-protocol.html#author (visited on 4th Apr. 2023).

[55] NTNU. 'Altium'. (), [Online]. Available: https://i.ntnu.no/wiki/-/wiki/English/Altium (visited on 26th Apr. 2023).

[56] NTNU. 'Ansys'. (), [Online]. Available: https://i.ntnu.no/wiki/-/wiki/English/ANSYS (visited on 7th May 2023).

[57] H. Zhang, 'What is the ltpowercad design tool and how is it different from ltspice?', *Analog Devices*, (visited on 5th Apr. 2023).

[58] GitHub. 'Github: Where the world builds software'. (), [Online]. Available: https://github.com (visited on 26th Feb. 2023).

[59] Zephyr. 'Zephyr project'. (), [Online]. Available: https://zephyrproject.org/wp-content/uploads/sites/38/2023/04/Zephyr-Overview-%202023Q1-Master.pdf (visited on 4th Apr. 2023).

[60] Nordic Semiconductor. 'Bare-metal vs rtos programming'. (), [Online]. Available: https://academy.nordicsemi.com/topic/bare-metal-vs-rtos-programming/ (visited on 10th May 2023).

[61] Zephyr. 'Kconfig'. (26th Apr. 2022), [Online]. Available: https://docs.zephyrproject.org/2.6.0/guides/kconfig/index.html (visited on 3rd May 2023).

[62] Zephyr. 'Application development'. (30th Sep. 2022), [Online]. Available: https://docs.zephyrproject.org/3.2.0/develop/application/index.html (visited on 3rd May 2023).

[63] Zephyr. 'Devicetree specification'. (30th Sep. 2022), [Online]. Available: https://docs.zephyrproject.org/3.2.0/build/dts/index.html (visited on 4th Apr. 2023).

[64] Zephyr. ‘Devicetree howtos’. (30th Sep. 2022), [Online]. Available: https://docs.zephyrproject.org/3.2.0/build/dts/howtos.html#set-devicetree-overlays (visited on 25th Mar. 2023).

[65] Microsoft. ‘Visual studio code’. (), [Online]. Available: https://code.visualstudio.com (visited on 23rd Mar. 2023).

[66] Nordic Semiconductor. ‘Nrf connect for desktop’. (2023), [Online]. Available: https://www.nordicsemi.com/Products/Development-tools/nrf-connect-for-desktop (visited on 4th Apr. 2023).

[67] Nordic Semiconductor. ‘Nrf connect for vs code’. (), [Online]. Available: https://www.nordicsemi.com/Products/Development-tools/nRF-Connect-for-VS-Code (visited on 27th Mar. 2023).

[68] Nordic Semiconductor. ‘Introduction to the zephyr rtos’. (24th Sep. 2020), [Online]. Available: https://webinars.nordicsemi.com/introduction-to-the-zephyr-rtos-3 (visited on 14th May 2023).

[69] Trimble. ‘3d design software’. (), [Online]. Available: https://www.sketchup.com/ (visited on 13th Apr. 2023).

[70] Zhejiang Flashforge 3D Technology Co., Ltd. ‘Flashprint 5’. (), [Online]. Available: https://www.flashforge.com/product-detail/40 (visited on 14th Apr. 2023).

[71] Analog Devices. ‘Ltm4622 datasheet’. (), [Online]. Available: https://www.analog.com/media/en/technical-documentation/data-sheets/ltm4622a.pdf (visited on 10th May 2023).

[72] Bosch Sensortec GmbH. ‘Bme680’. (18th Aug. 2022), [Online]. Available: https://www.bosch-sensortec.com/media/boschsensortec/downloads/datasheets/bst-bme680-ds001.pdf (visited on 1st Apr. 2023).

[73] Bosch Sensortec. ‘Bme688’. (), [Online]. Available: https://no.mouser.com/datasheet/2/783/bst_bme688_fl000-2307034.pdf (visited on 7th Apr. 2023).

[74] Sensirion. ‘Datasheet sht4x’. (), [Online]. Available: https://www.mouser.com/datasheet/2/682/Datasheet_SHT4x-1917879.pdf (visited on 3rd May 2023).

[75] Analog devices. ‘Adxl372’. (), [Online]. Available: https://www.analog.com/en/products/adxl372.html#product-overview (visited on 6th Apr. 2023).

[76] Analog devices. ‘Adxl362’. (), [Online]. Available: https://www.analog.com/en/products/adxl362.html#product-overview (visited on 6th Apr. 2023).

[77] Nordic Semiconductors. ‘Nrf5340’. (7th Mar. 2023), [Online]. Available: https://infocenter.nordicsemi.com/topic/struct_nrf53/struct/nrf5340.html (visited on 7th Mar. 2023).

[78] Nordic Semiconductors. ‘Antenna requirements’. (), [Online]. Available: https://infocenter.nordicsemi.com/topic/nwp_033/WP/nwp_033/nwp_033_antenna_req.html (visited on 10th May 2023).

[79] Ignion. ‘A standard antenna solution for mobile frequency bands’. (1st Nov. 2017), [Online]. Available: https://no.mouser.com/datasheet/2/1029/UM_FR01_S4_210_1-1863540.pdf (visited on 10th May 2023).

[80] Ignion. ‘Antenna intelligence cloud’. (), [Online]. Available: https://ignion.io/antenna-intelligence/ (visited on 10th May 2023).

[81] Ignion. ‘One mxtendtm’. (1st Jan. 2022), [Online]. Available: https://ignion.io/files/DS_NN02-201.pdf (visited on 10th May 2023).

[82] Ignion. ‘Mini-antenna for any cellular application’. (1st Jan. 2021), [Online]. Available: https://ignion.io/files/AN_NN02-201-2G_3G.pdf (visited on 10th May 2023).

[83] HBR. ‘Norwegian 5g’. (), [Online]. Available: https://halberdbastion.com/intelligence/countries-nations/norway (visited on 25th Apr. 2023).

[84] Johanson Technologies inc. ‘2450at18d0100’. (2021), [Online]. Available: https://www.johansontechnology.com/datasheets/2450AT18D0100/2450AT18D0100.pdf (visited on 5th Apr. 2023).

[85] Taoglas. ‘Aggbp.sl.18a active gnss patch’. (), [Online]. Available: https://cdn.taoglas.com/datasheets/AGGBP.SL.18A.07.0060A.pdf (visited on 15th Apr. 2023).

[86] Nordic Semiconductor. 'Nrf52832 product specification'. (24th Apr. 2023), [Online]. Available: https://infocenter.nordicsemi.com/topic/struct_nrf52/struct/nrf52832_ps.html (visited on 1st May 2023).

[87] Nordic Semiconductor. 'Coap protocol'. (2023), [Online]. Available: https://academy.nordicsemi.com/topic/lesson-5-coap-protocol/ (visited on 16th Apr. 2023).

[88] Eclipse. 'Eclipse californium'. (), [Online]. Available: https://www.eclipse.org/californium/ (visited on 13th May 2023).

[89] Nordic Semiconductor. 'Bluetooth le advertising - exercise 2'. (), [Online]. Available: https://academy.nordicsemi.com/topic/blefund-lesson-2-exercise-2/ (visited on 21st May 2023).

[90] Digikey. 'Understanding the advantages and disadvantages of linear regulators'. (7th Apr. 2023), [Online]. Available: https://www.digikey.no/en/articles/understanding-the-advantages-and-disadvantages-of-linear-regulators (visited on 7th Apr. 2023).

[91] J. G. Balchen, T. Andresen and B. A. Foss, *Reguleringsteknikk*. NTNU Grafisk Senter, 2016.

[92] Texas Instruments. 'Common mistakes in dc/dc converters and how to fix them'. (2018), [Online]. Available: https://www.ti.com/seclit/ml/slup384/slup384.pdf (visited on 19th May 2023).

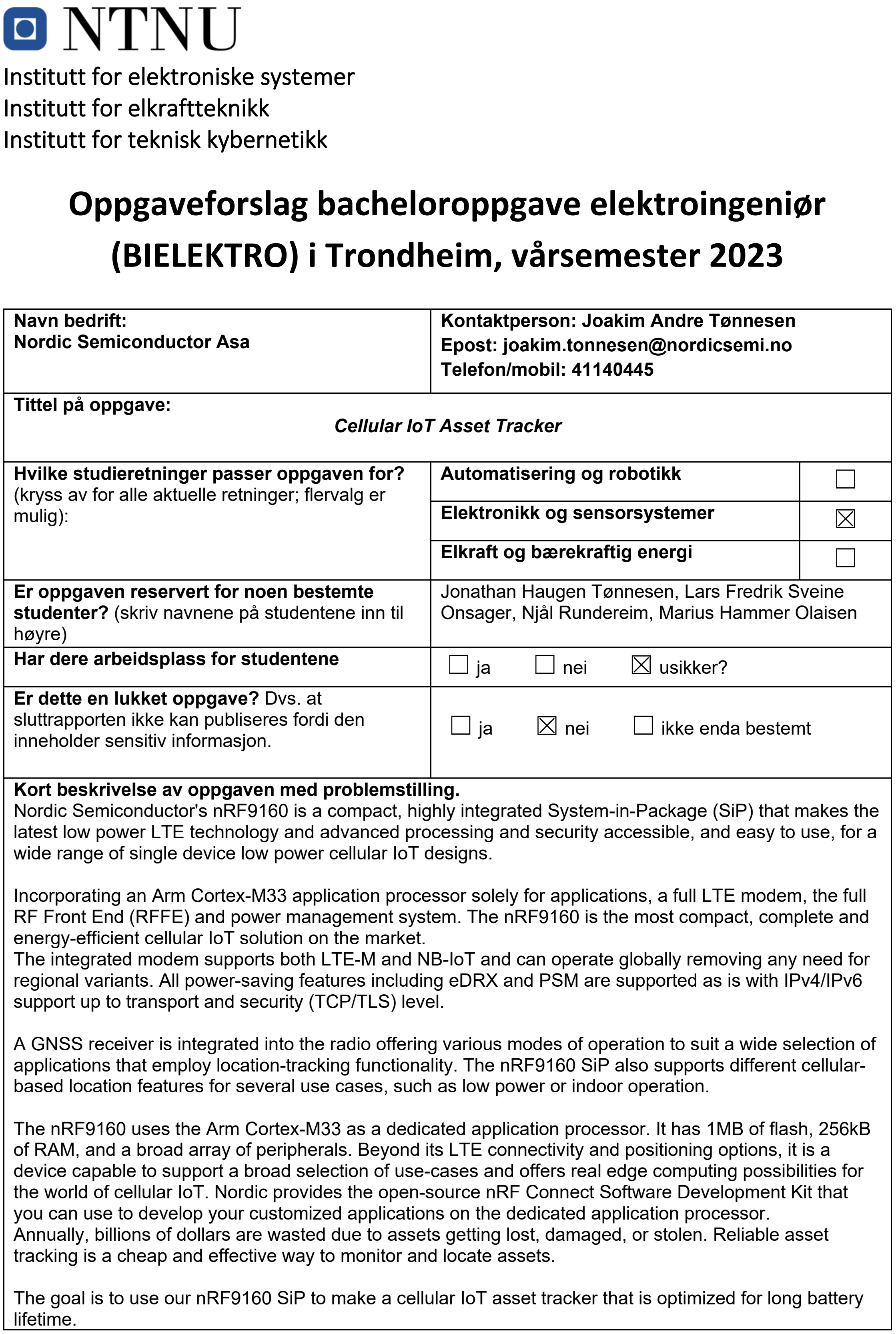
[93] Nordic Semiconductor. 'Nordic devzone'. (), [Online]. Available: https://devzone.nordicsemi.com (visited on 15th May 2023).

[94] Analog Devices. 'Ltc3549 datasheet'. (), [Online]. Available: https://www.analog.com/media/en/technical-documentation/data-sheets/3549f.pdf (visited on 5th Apr. 2023).

[95] Silicon Labs. 'Usbxpress family cp2102n data sheet'. (7th Mar. 2023), [Online]. Available: https://www.silabs.com/documents/public/data-sheets/cp2102n-datasheet.pdf (visited on 7th Mar. 2023).

[96] Nordic Semiconductors. 'Nrf5340'. (7th Mar. 2023), [Online]. Available: https://infocenter.nordicsemi.com/topic/struct_nrf53/struct/nrf5340.html (visited on 7th May 2023).

A Bachelor Assignment



NTNU

Institutt for elektroniske systemer
Institutt for elkraftteknikk
Institutt for teknisk kybernetikk

Oppgaveforslag bacheloroppgave elektroingeniør (BIELEKTRO) i Trondheim, vårsemester 2023

Navn bedrift: **Nordic Semiconductor Asa**	**Kontaktperson: Joakim Andre Tønnesen** **Epost: joakim.tonnesen@nordicsemi.no** **Telefon/mobil: 41140445**	
Tittel på oppgave: ***Cellular IoT Asset Tracker***		
Hvilke studieretninger passer oppgaven for? (kryss av for alle aktuelle retninger; flervalg er mulig):	**Automatisering og robotikk**	☐
	Elektronikk og sensorsystemer	☒
	Elkraft og bærekraftig energi	☐
Er oppgaven reservert for noen bestemte studenter? (skriv navnene på studentene inn til høyre)	Jonathan Haugen Tønnesen, Lars Fredrik Sveine Onsager, Njål Rundereim, Marius Hammer Olaisen	
Har dere arbeidsplass for studentene	☐ ja ☐ nei ☒ usikker?	
Er dette en lukket oppgave? Dvs. at sluttrapporten ikke kan publiseres fordi den inneholder sensitiv informasjon.	☐ ja ☒ nei ☐ ikke enda bestemt	

Kort beskrivelse av oppgaven med problemstilling.
Nordic Semiconductor's nRF9160 is a compact, highly integrated System-in-Package (SiP) that makes the latest low power LTE technology and advanced processing and security accessible, and easy to use, for a wide range of single device low power cellular IoT designs.

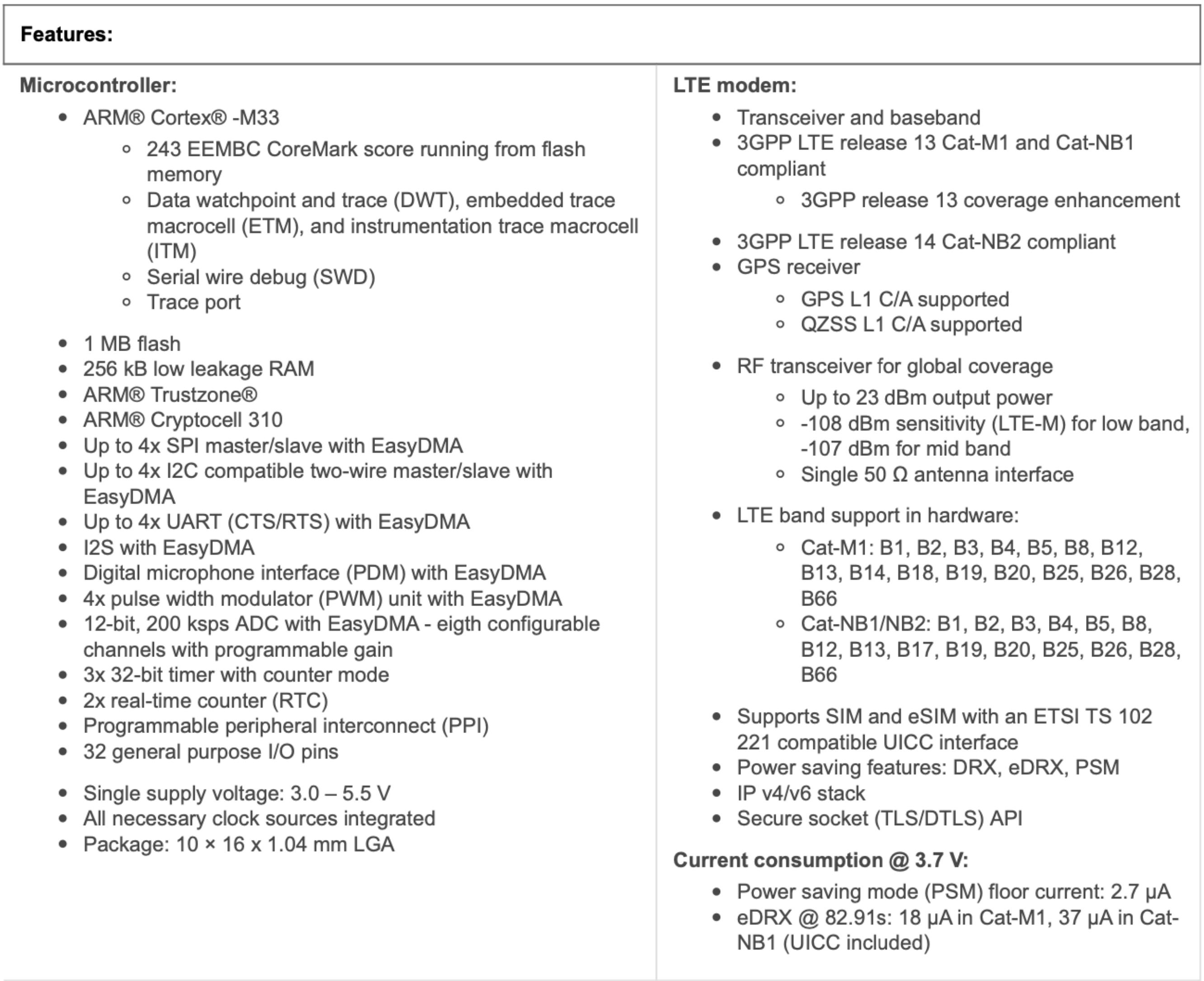
Incorporating an Arm Cortex-M33 application processor solely for applications, a full LTE modem, the full RF Front End (RFFE) and power management system. The nRF9160 is the most compact, complete and energy-efficient cellular IoT solution on the market.
The integrated modem supports both LTE-M and NB-IoT and can operate globally removing any need for regional variants. All power-saving features including eDRX and PSM are supported as is with IPv4/IPv6 support up to transport and security (TCP/TLS) level.

A GNSS receiver is integrated into the radio offering various modes of operation to suit a wide selection of applications that employ location-tracking functionality. The nRF9160 SiP also supports different cellular-based location features for several use cases, such as low power or indoor operation.

The nRF9160 uses the Arm Cortex-M33 as a dedicated application processor. It has 1MB of flash, 256kB of RAM, and a broad array of peripherals. Beyond its LTE connectivity and positioning options, it is a device capable to support a broad selection of use-cases and offers real edge computing possibilities for the world of cellular IoT. Nordic provides the open-source nRF Connect Software Development Kit that you can use to develop your customized applications on the dedicated application processor.
Annually, billions of dollars are wasted due to assets getting lost, damaged, or stolen. Reliable asset tracking is a cheap and effective way to monitor and locate assets.

The goal is to use our nRF9160 SiP to make a cellular IoT asset tracker that is optimized for long battery lifetime.

B nRF9160 Features Summary



Features:

Microcontroller:

- ARM® Cortex® -M33
  - 243 EEMBC CoreMark score running from flash memory
  - Data watchpoint and trace (DWT), embedded trace macrocell (ETM), and instrumentation trace macrocell (ITM)
  - Serial wire debug (SWD)
  - Trace port
- 1 MB flash
- 256 kB low leakage RAM
- ARM® Trustzone®
- ARM® Cryptocell 310
- Up to 4x SPI master/slave with EasyDMA
- Up to 4x I2C compatible two-wire master/slave with EasyDMA
- Up to 4x UART (CTS/RTS) with EasyDMA
- I2S with EasyDMA
- Digital microphone interface (PDM) with EasyDMA
- 4x pulse width modulator (PWM) unit with EasyDMA
- 12-bit, 200 ksps ADC with EasyDMA - eigth configurable channels with programmable gain
- 3x 32-bit timer with counter mode
- 2x real-time counter (RTC)
- Programmable peripheral interconnect (PPI)
- 32 general purpose I/O pins

- Single supply voltage: 3.0 – 5.5 V
- All necessary clock sources integrated
- Package: 10 × 16 x 1.04 mm LGA

LTE modem:

- Transceiver and baseband
- 3GPP LTE release 13 Cat-M1 and Cat-NB1 compliant
  - 3GPP release 13 coverage enhancement
- 3GPP LTE release 14 Cat-NB2 compliant
- GPS receiver
  - GPS L1 C/A supported
  - QZSS L1 C/A supported
- RF transceiver for global coverage
  - Up to 23 dBm output power
  - -108 dBm sensitivity (LTE-M) for low band, -107 dBm for mid band
  - Single 50 Ω antenna interface
- LTE band support in hardware:
  - Cat-M1: B1, B2, B3, B4, B5, B8, B12, B13, B14, B18, B19, B20, B25, B26, B28, B66
  - Cat-NB1/NB2: B1, B2, B3, B4, B5, B8, B12, B13, B17, B19, B20, B25, B26, B28, B66
- Supports SIM and eSIM with an ETSI TS 102 221 compatible UICC interface
- Power saving features: DRX, eDRX, PSM
- IP v4/v6 stack
- Secure socket (TLS/DTLS) API

Current consumption @ 3.7 V:

- Power saving mode (PSM) floor current: 2.7 µA
- eDRX @ 82.91s: 18 µA in Cat-M1, 37 µA in Cat-NB1 (UICC included)

C Schematics

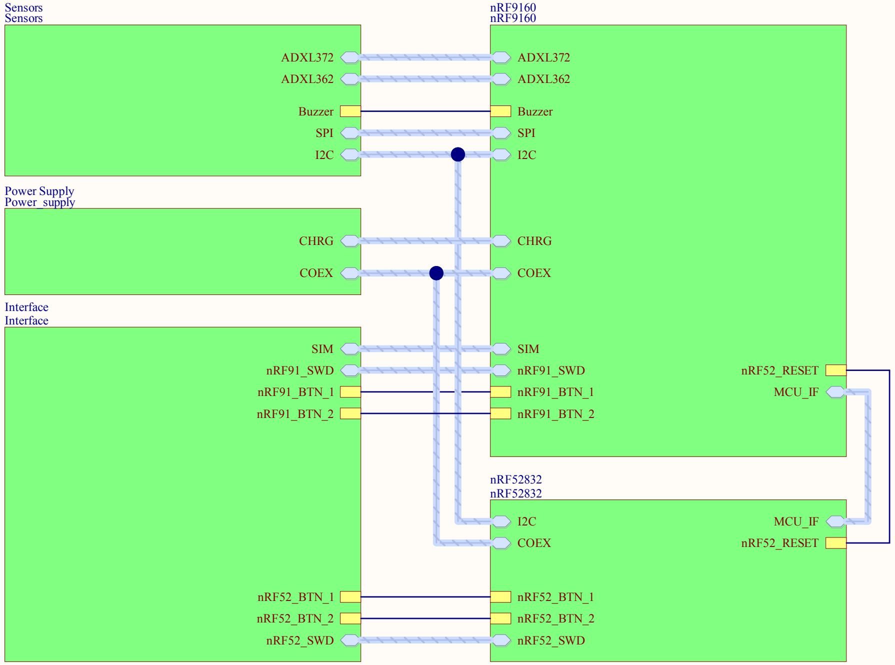
Main unit

1. Overview
2. Power supply
3. Sensors
4. Interface
5. nRF9160
6. nRF52842

AUX Unit

Bachelor thesis 2023

Table of Content



Project
Main_unit.PrjPcb

A3

Name
Main_sheet.SchDoc

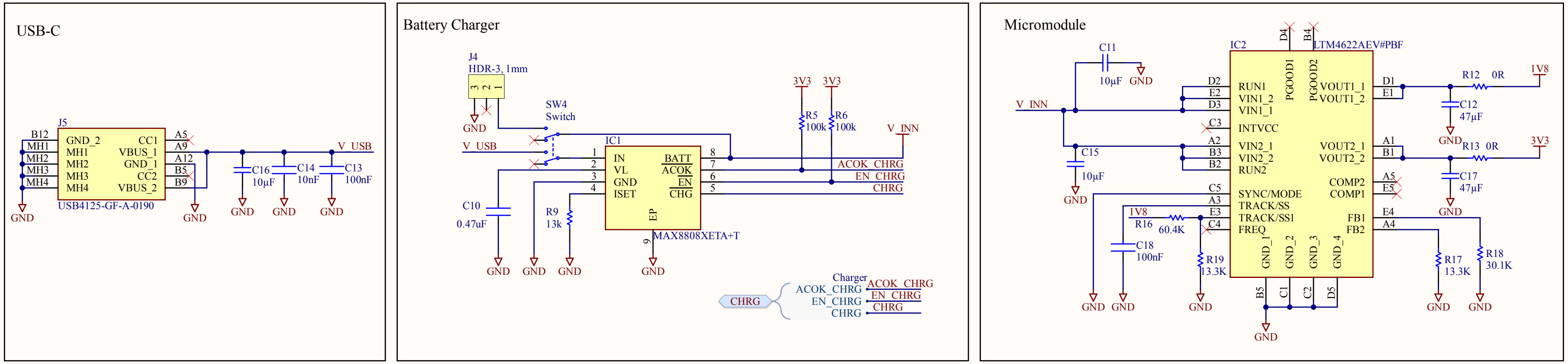
Sheet
1 / 6

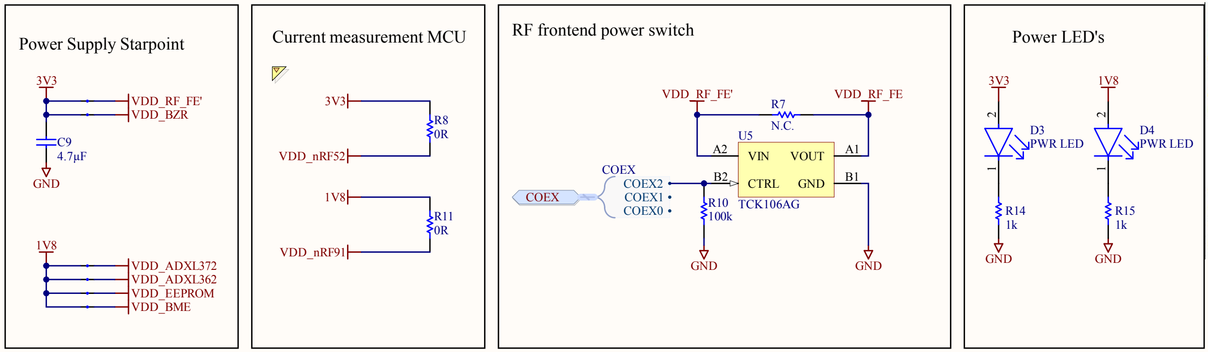
Rev
Not in

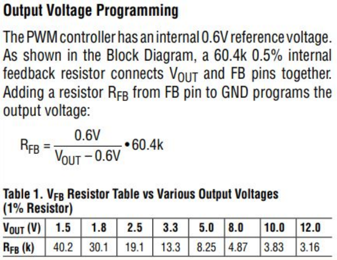
Designer

Date
25.04.2023

Power supply







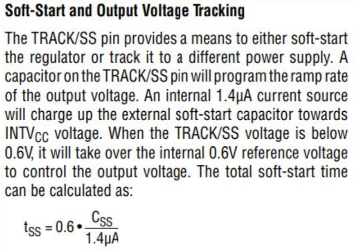
Output Voltage Programming

The PWM controller has an internal 0.6V reference voltage. As shown in the Block Diagram, a 60.4k 0.5% internal feedback resistor connects V_{OUT} and FB pins together. Adding a resistor R_{FB} from FB pin to GND programs the output voltage:

$$R_{FB} = \frac{0.6V}{V_{OUT} - 0.6V} \bullet 60.4k$$

Table 1. V_{FB} Resistor Table vs Various Output Voltages (1% Resistor)

V_{OUT} (V)	1.5	1.8	2.5	3.3	5.0	8.0	10.0	12.0
R_{FB} (k)	40.2	30.1	19.1	13.3	8.25	4.87	3.83	3.16



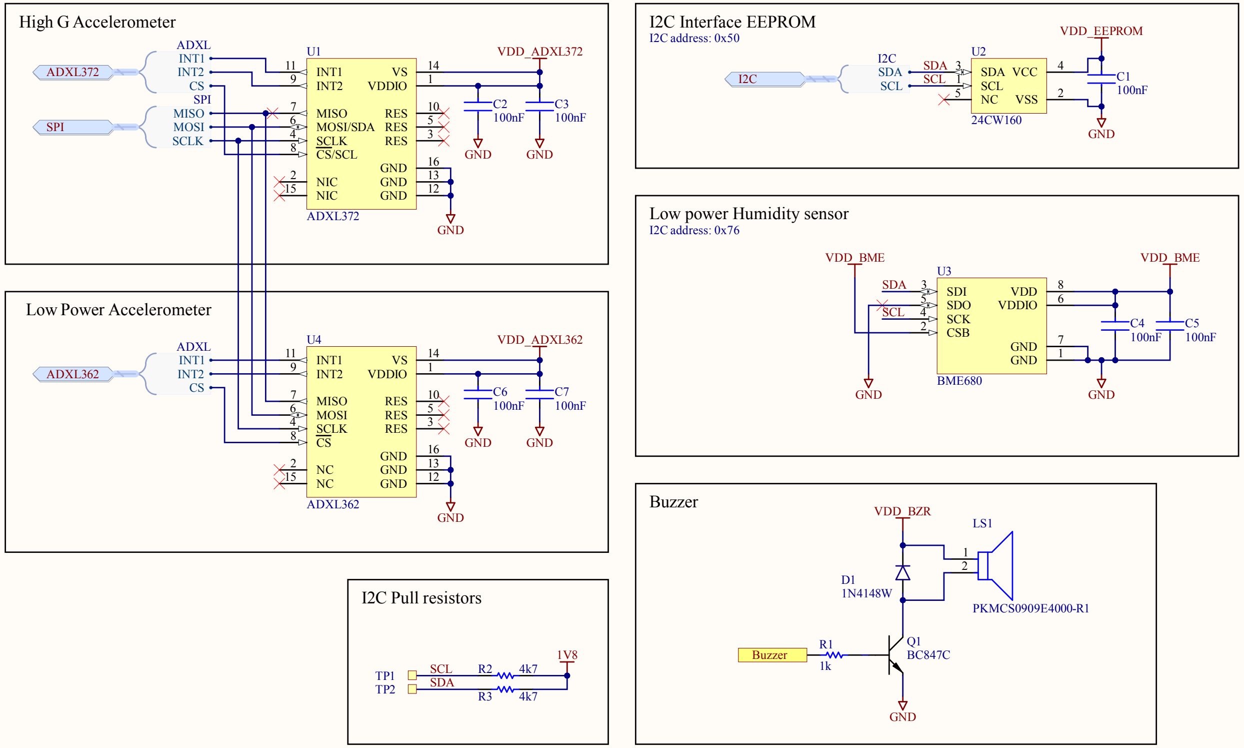
Soft-Start and Output Voltage Tracking

The TRACK/SS pin provides a means to either soft-start the regulator or track it to a different power supply. A capacitor on the TRACK/SS pin will program the ramp rate of the output voltage. An internal 1.4µA current source will charge up the external soft-start capacitor towards $INTV_{CC}$ voltage. When the TRACK/SS voltage is below 0.6V, it will take over the internal 0.6V reference voltage to control the output voltage. The total soft-start time can be calculated as:

$$t_{SS} = 0.6 \bullet \frac{C_{SS}}{1.4\mu A}$$

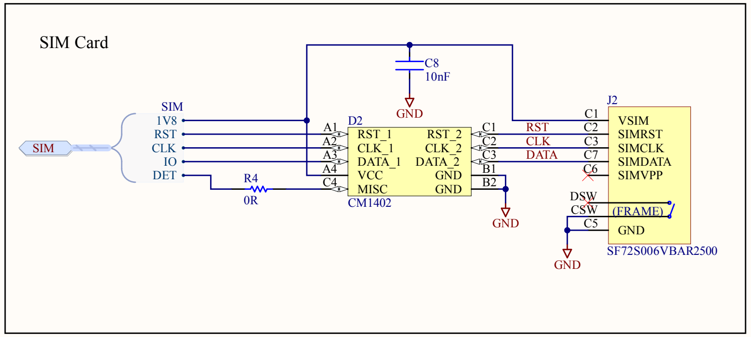
Project Main_unit.PrjPcb	A3
Name Power_supply.SchDoc	Sheet 2 / 6
Rev Not in	Designer	Date 20.04.2023

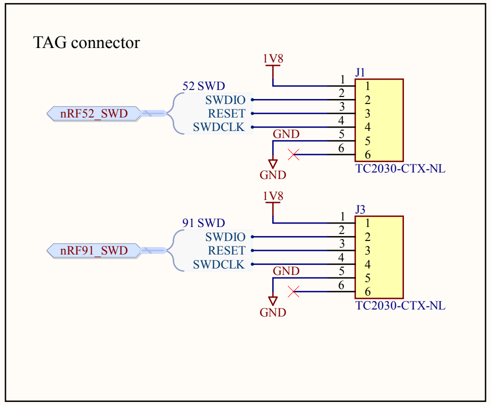
Sensors

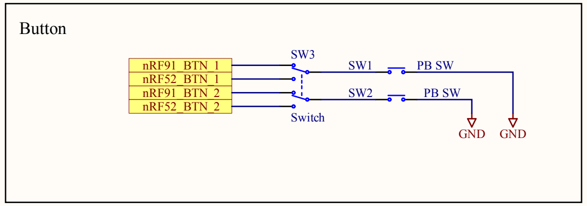


Project Main_unit.PrjPcb		A3
Name Sensors.SchDoc	Sheet 3 / 6	
Rev Not in	Designer	Date 20.04.2023

Interface







Project
Main_unit.PrjPcb

A3

Name
Interface.SchDoc

Sheet
4 / 6

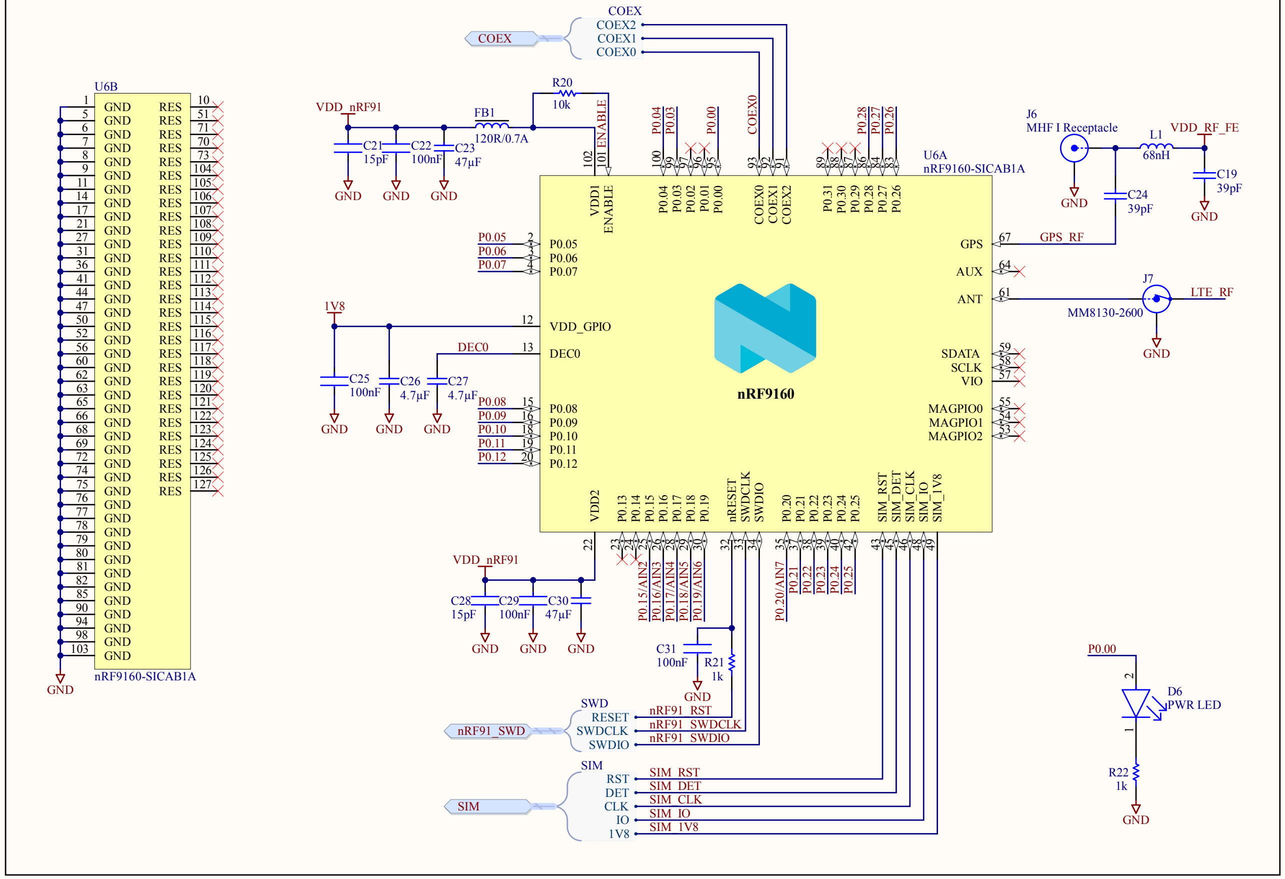
Rev
Not in

Designer

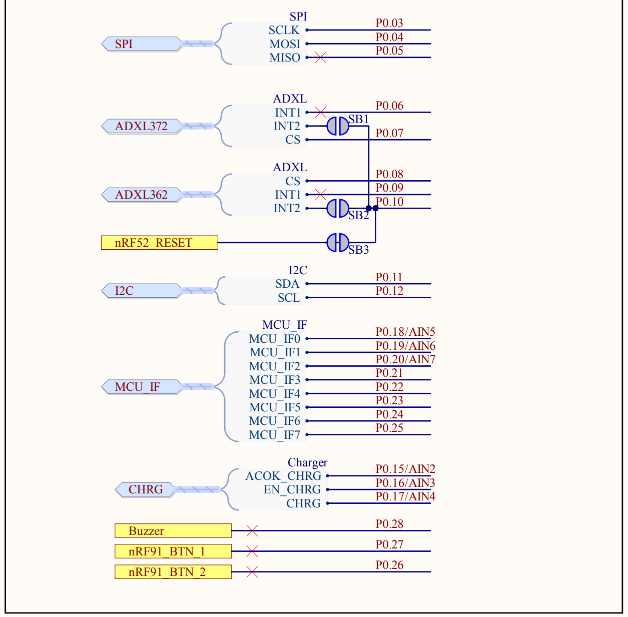
Date
24.04.2023

nRF9160

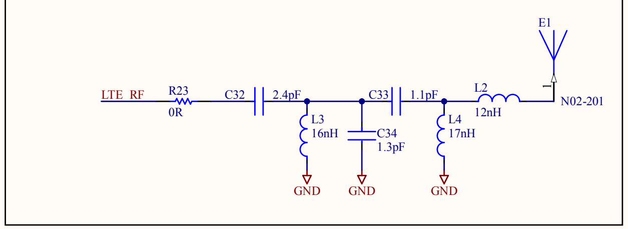
nRF9160



In / Out

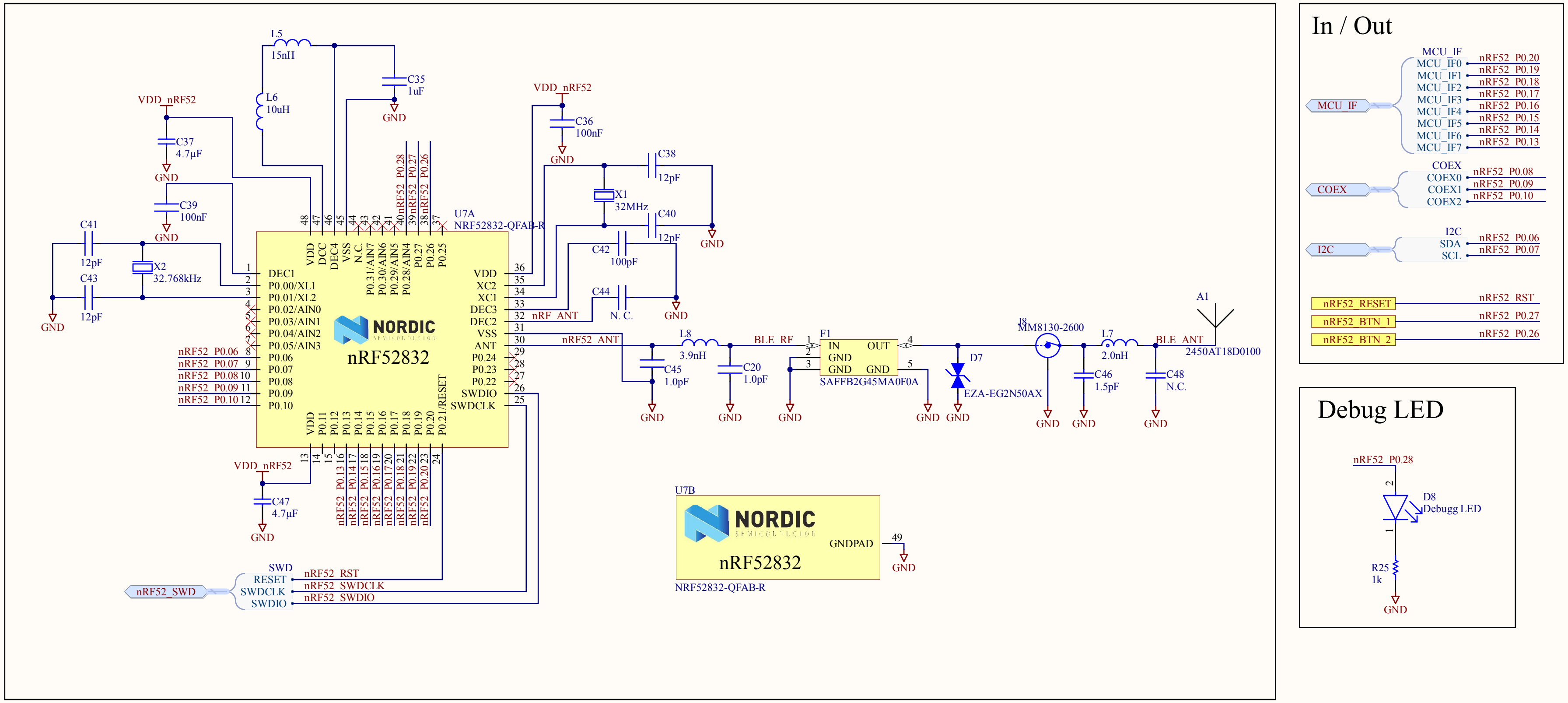


LTE antenna and matching network



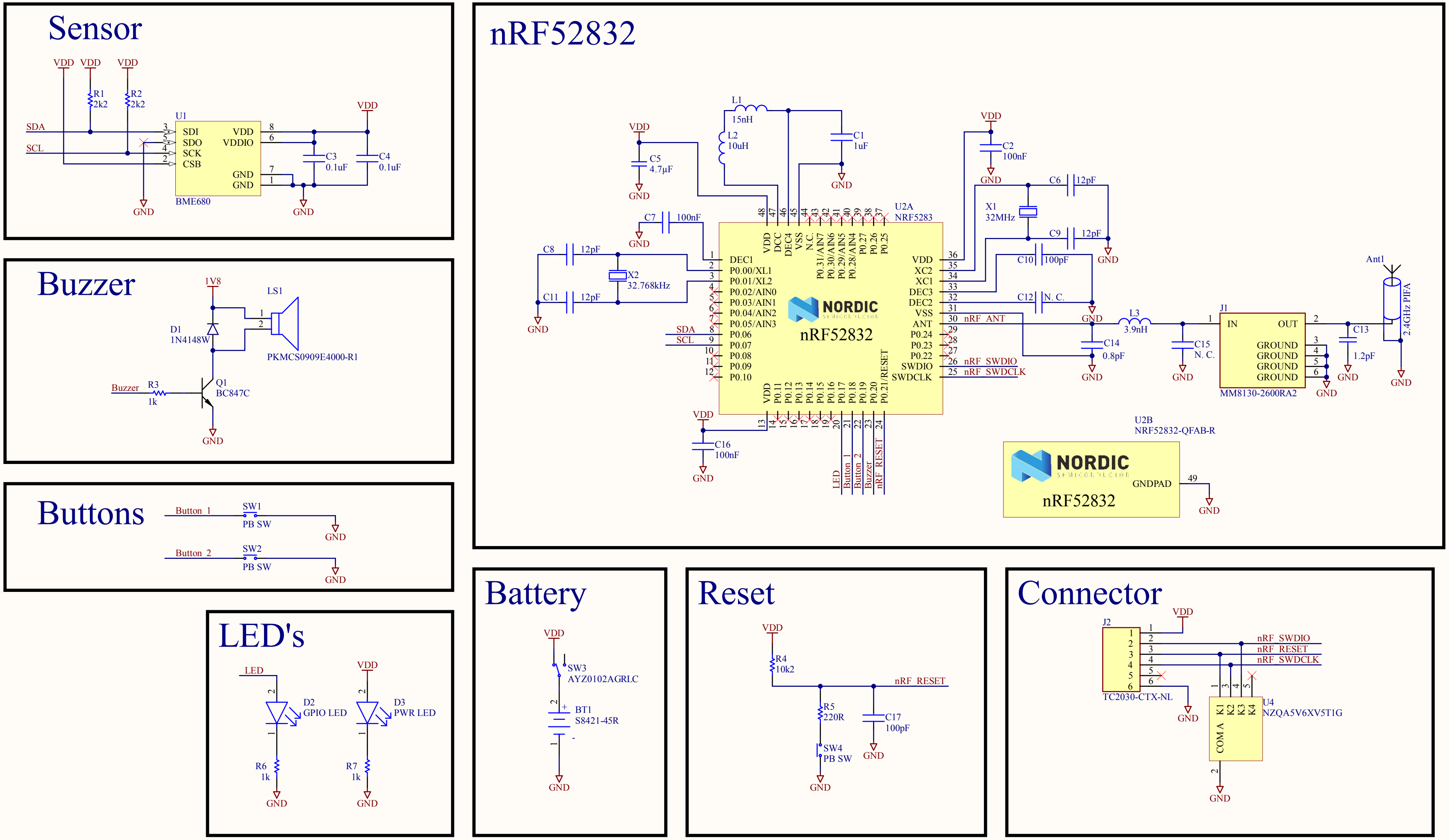
Project		A3
Main_unit.PrjPcb		
Name	Sheet	
nRF9160.SchDoc	5 / 6	
Rev	Designer	Date
Not in		22.04.2023

nRF52832



Project		A3
Main_unit.PrjPcb		
Name	Sheet	
nRF52832.SchDoc	6 / 6	
Rev	Designer	Date
Not in		20.04.2023

Auxiliary Unit



Project:
Auxiliary_Unit.PrjPcb

Title:
Main.SchDoc

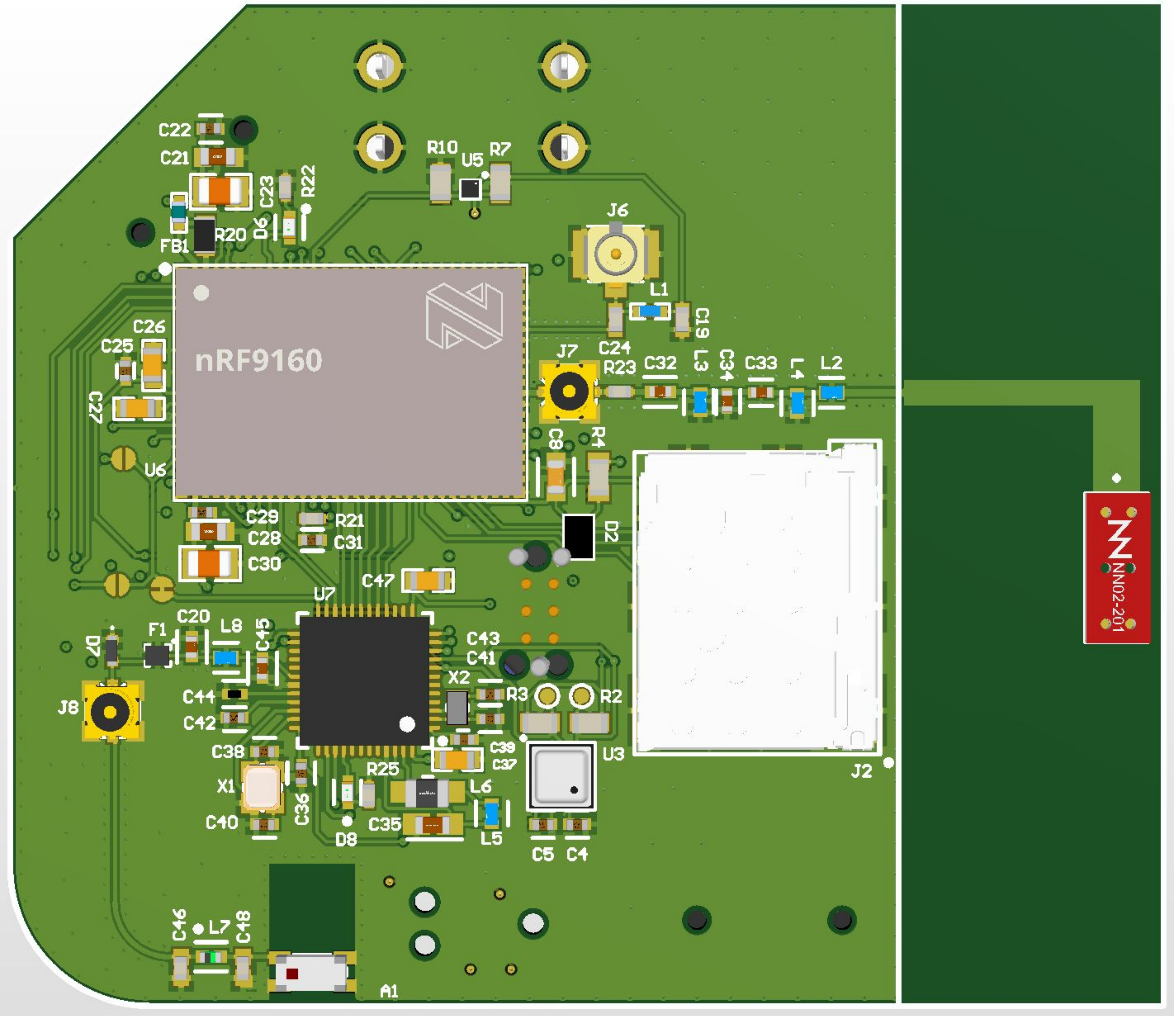
Sheet number:
1 out of

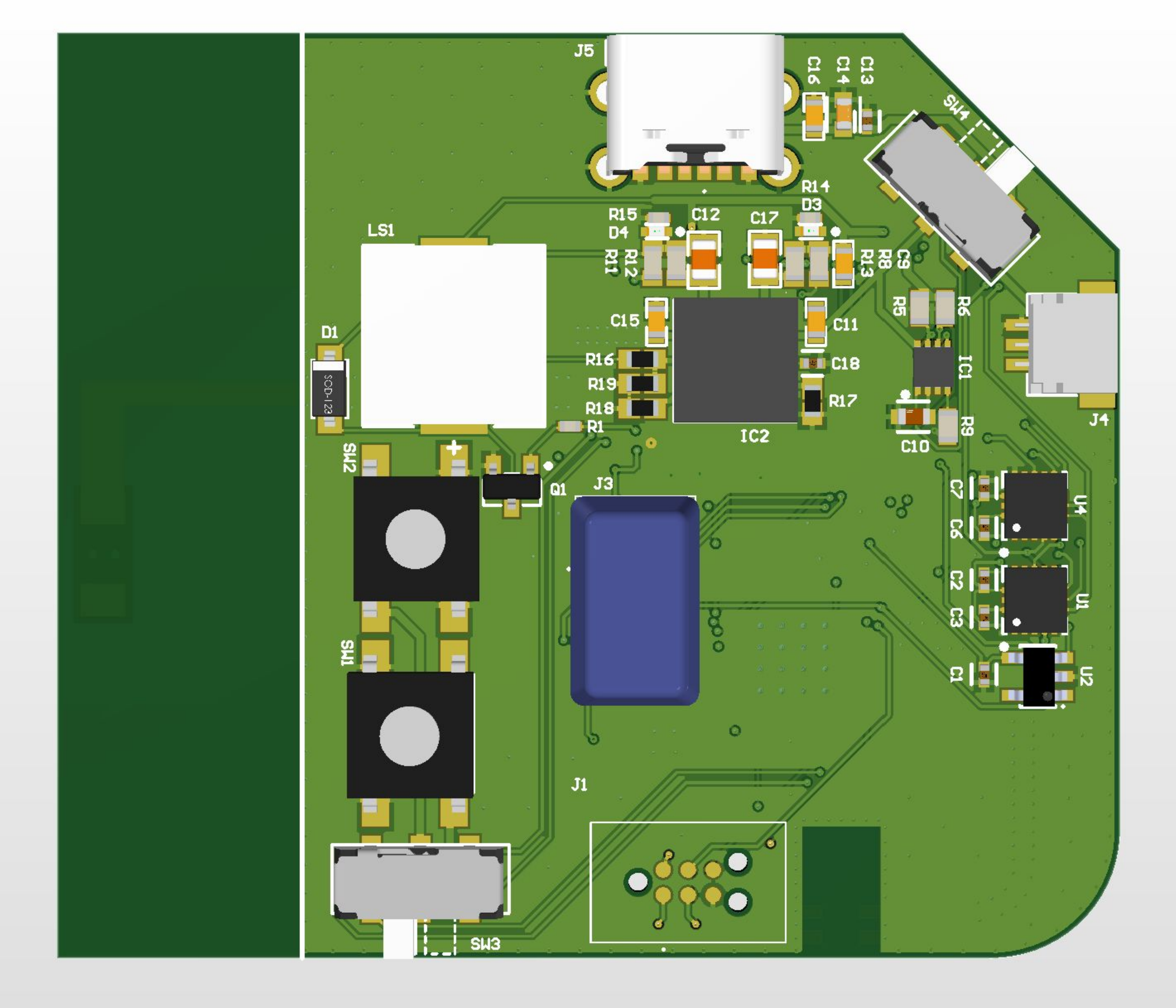
Revision:

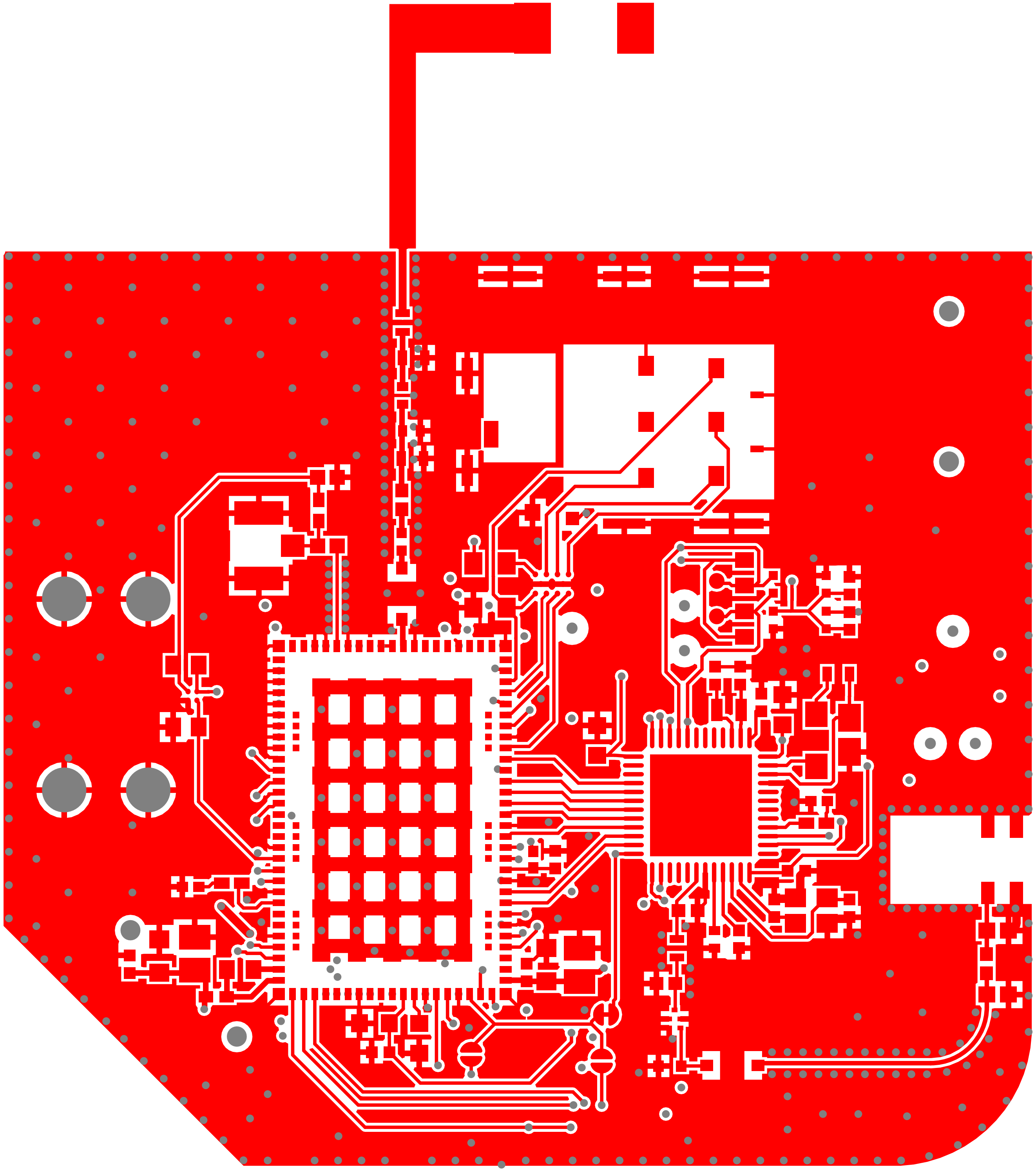
Designer:
Njål Rundereim

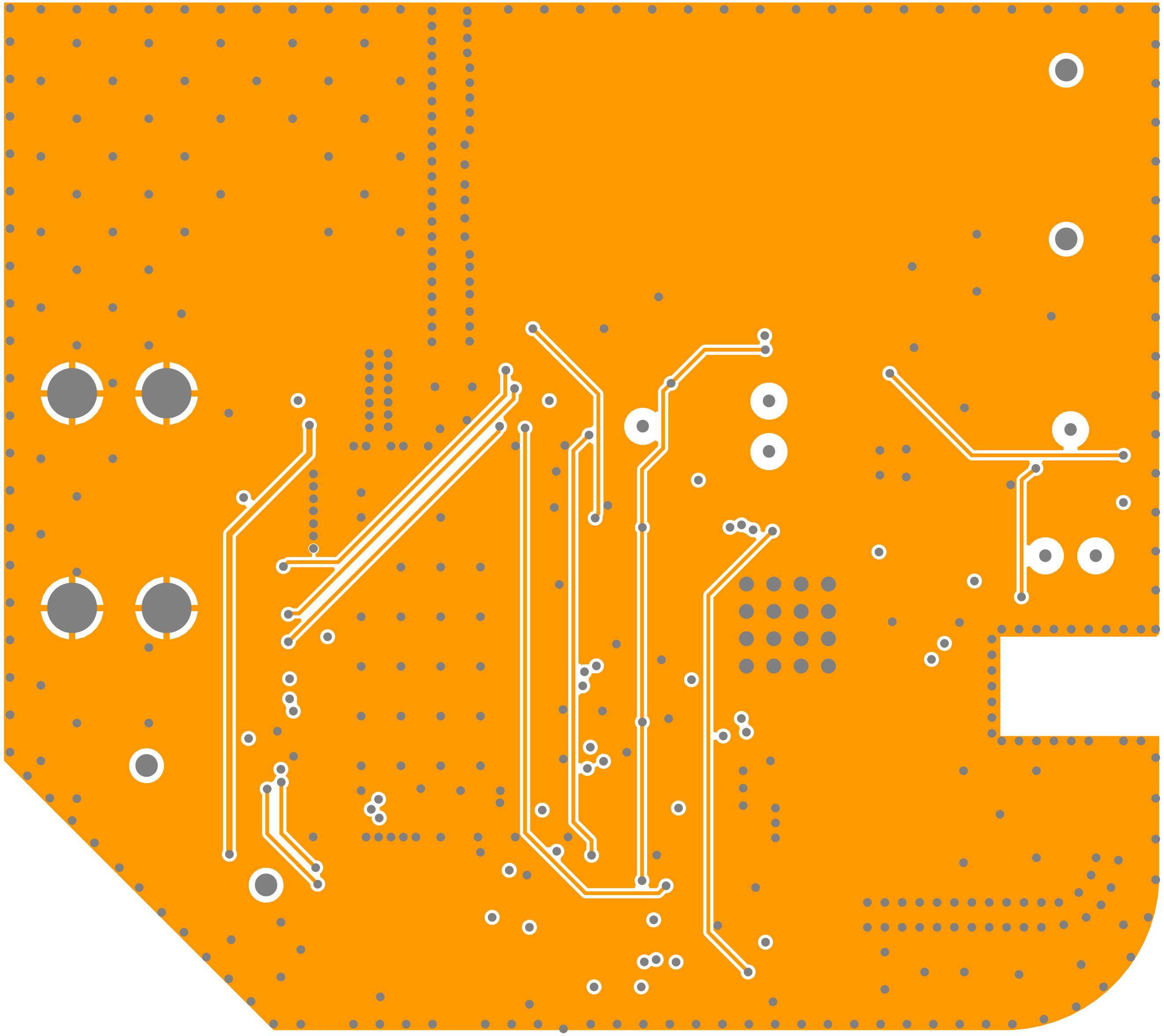
Date:
16.05.2023

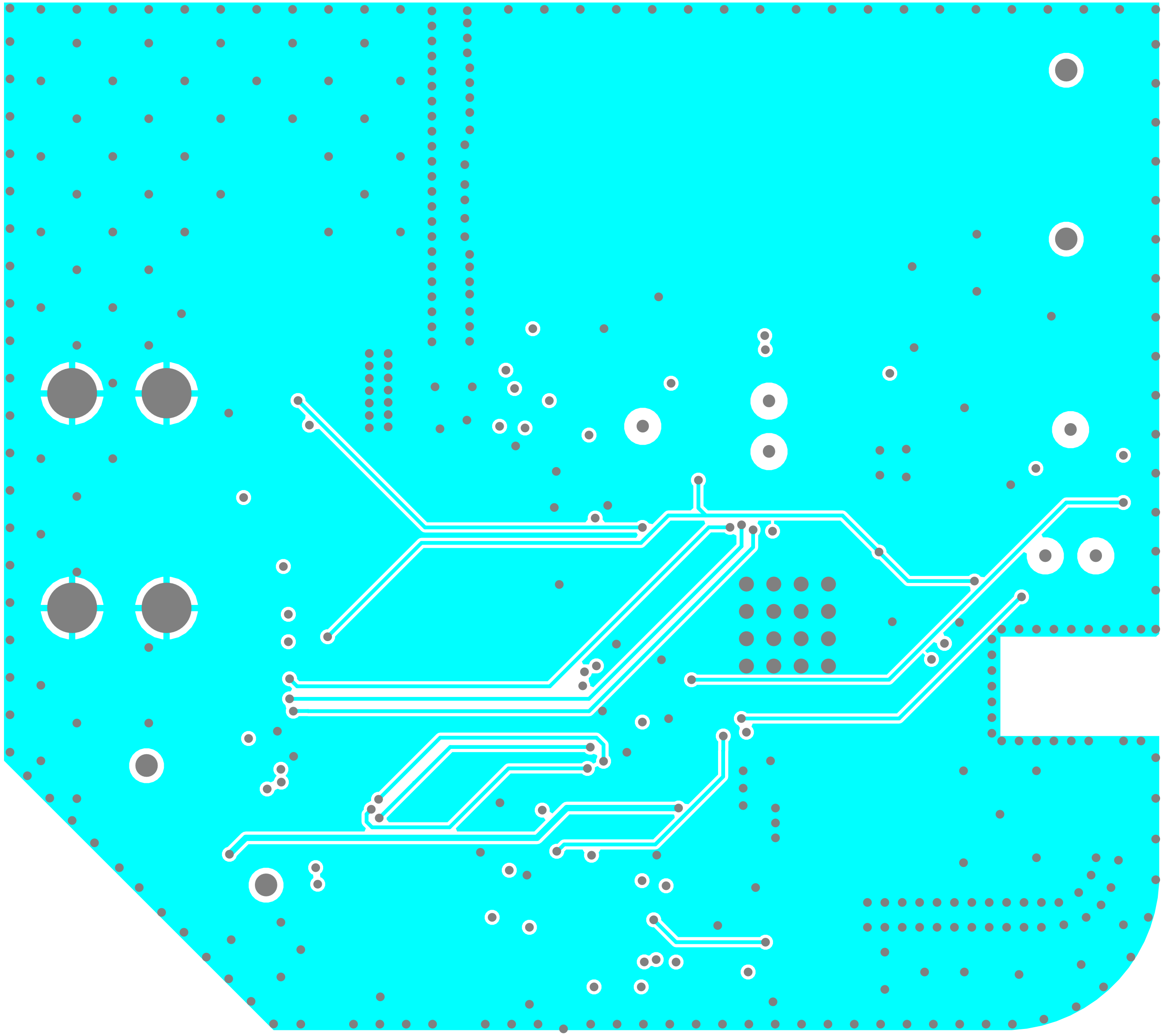
D Circuit board layout

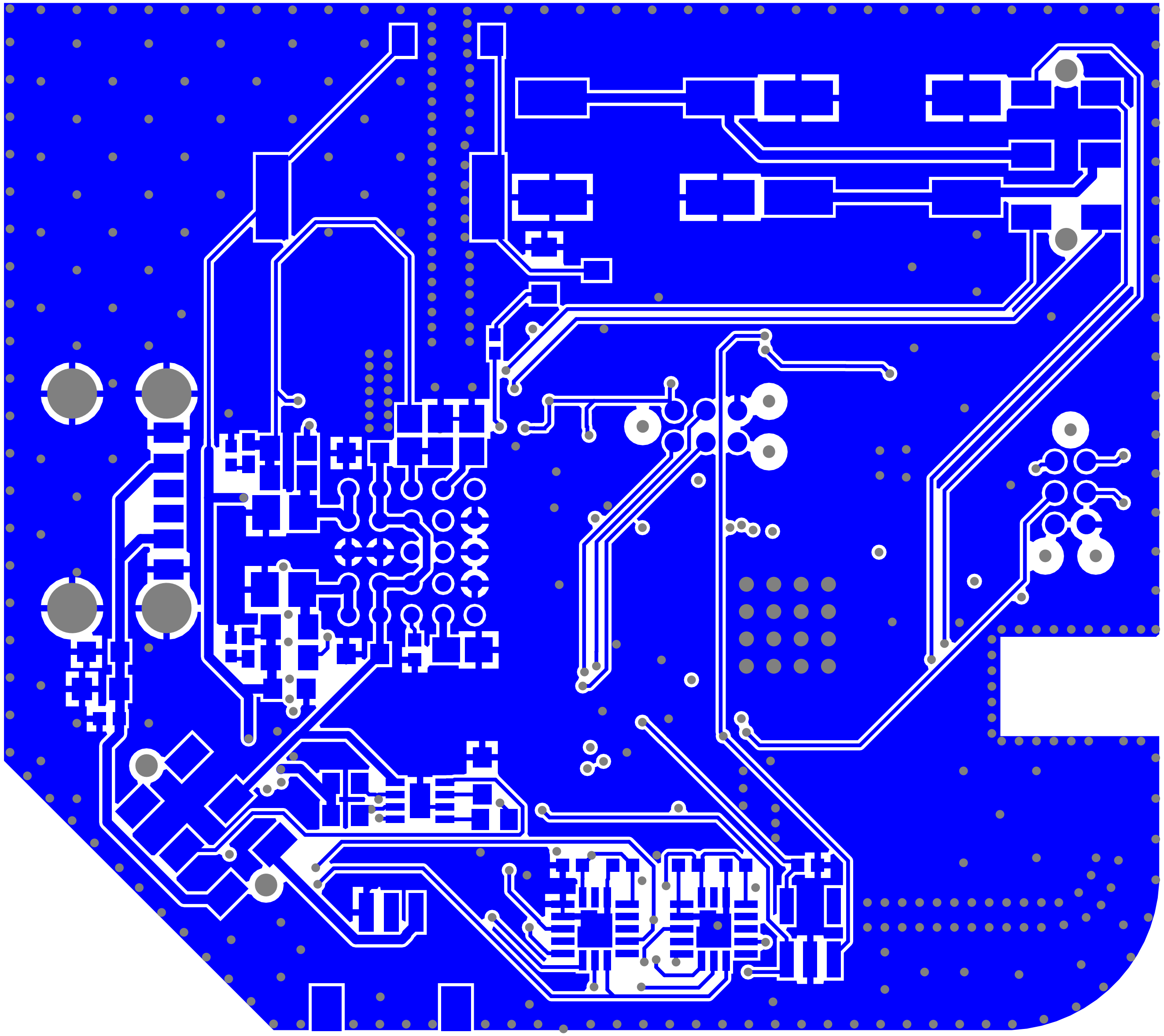

C22
C21
C23
R22
R10
U5
R7
FB1
R20
D6
J6
nRF9160
L1
C19
C26
C25
C27
J7
C24
R23
C32
L3
C34
C33
L4
L2
C8
R4
U6
C29
R21
C28
C31
C30
D2
C47
U7
C20
F1
L8
C45
D7
C43
C41
X2
J8
C44
C42
R3
R2
C38
C39
C37
U3
J2
X1
R25
L6
C36
C40
D8
C35
L5
C5
C4
C46
L7
C48
A1
NN02-201

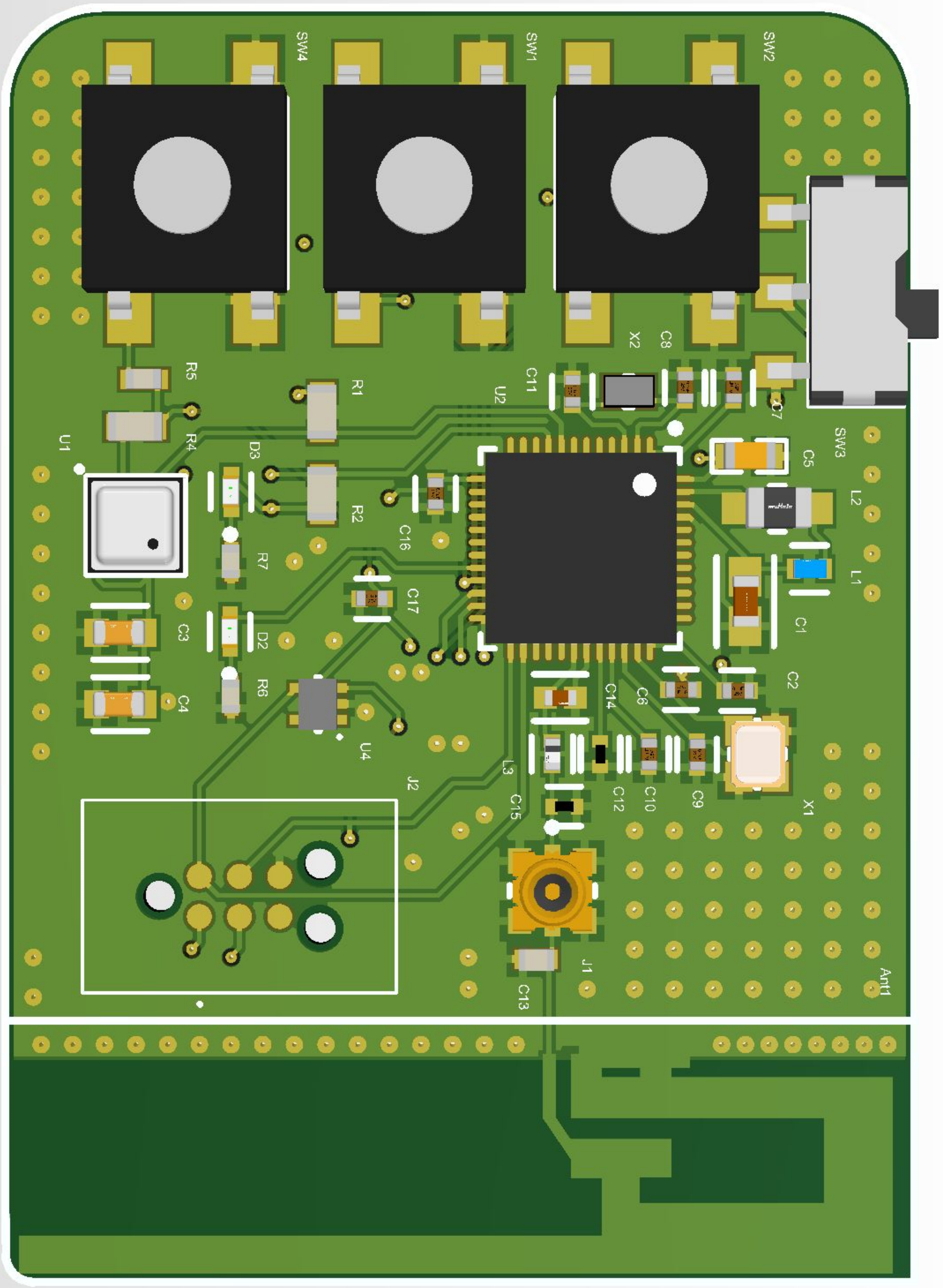

J5
C16
C14
C13
SW4
R14
D3
R15
D4
C12
C17
R11
R12
R13
R8
C9
LS1
C15
C11
R5
R6
D1
R16
R19
R18
R1
C18
R17
IC1
IC2
C10
R9
J4
SW2
Q1
J3
C7
C6
U4
C2
C3
U1
SW1
C1
U2
J1
SW3

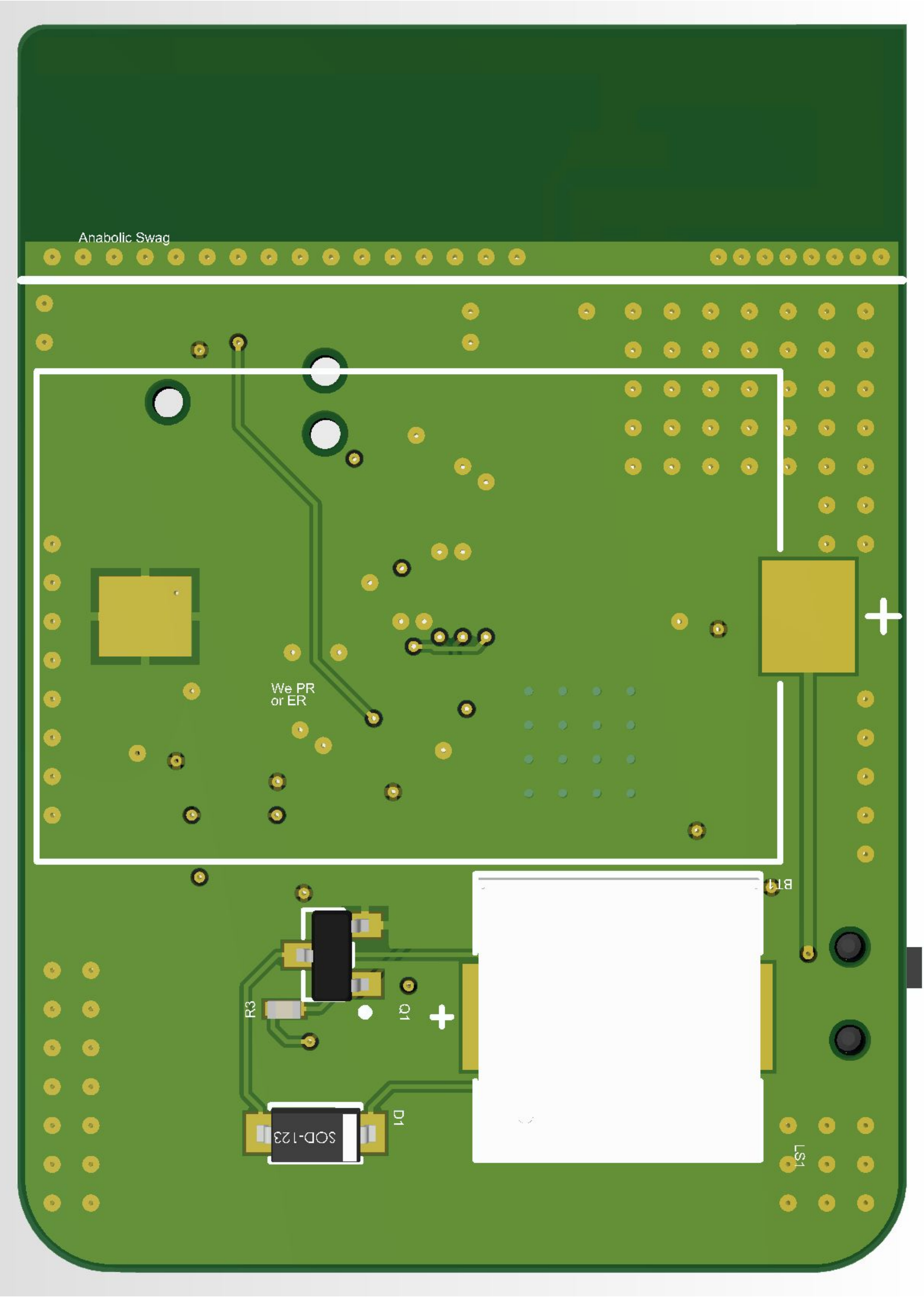


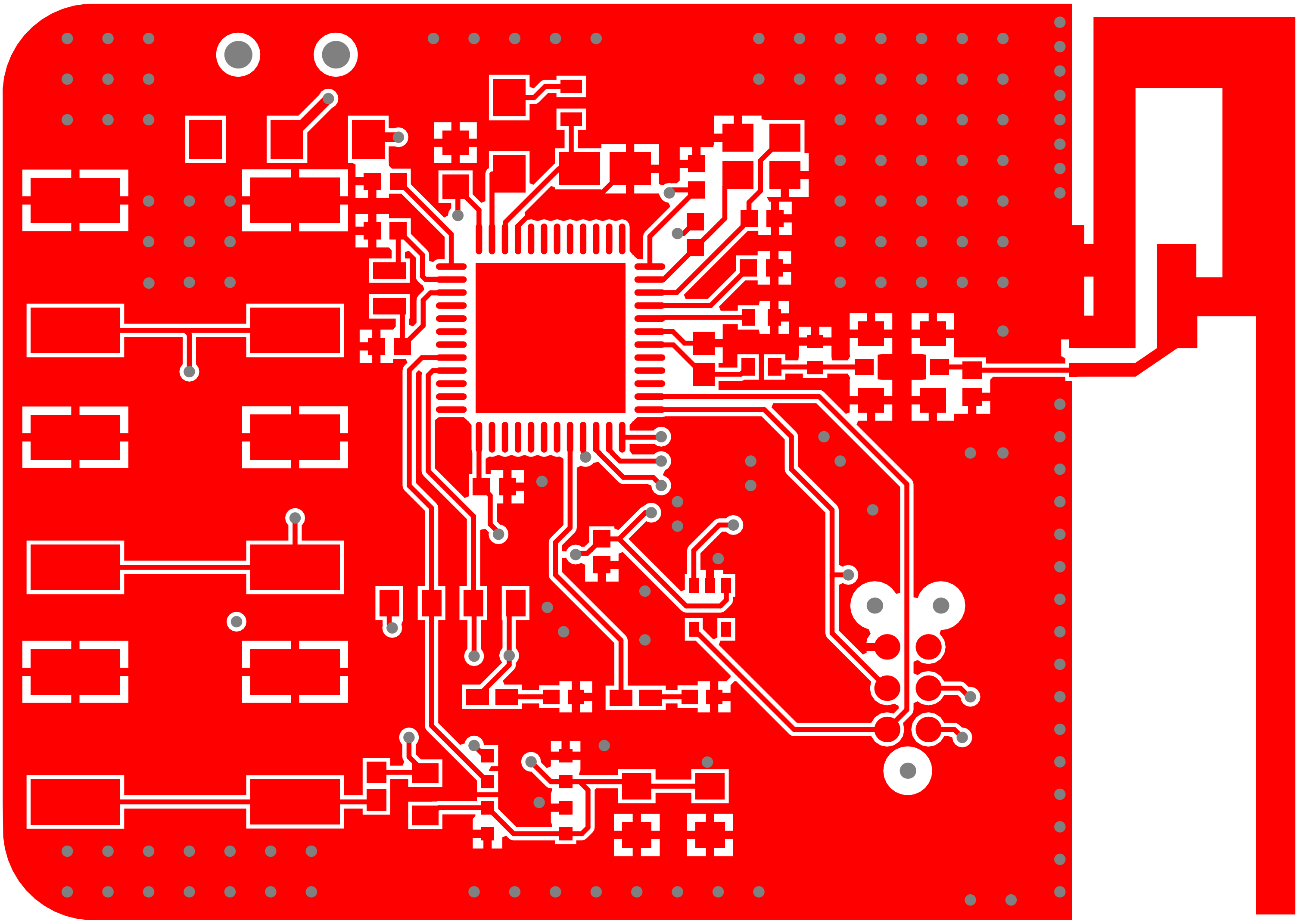


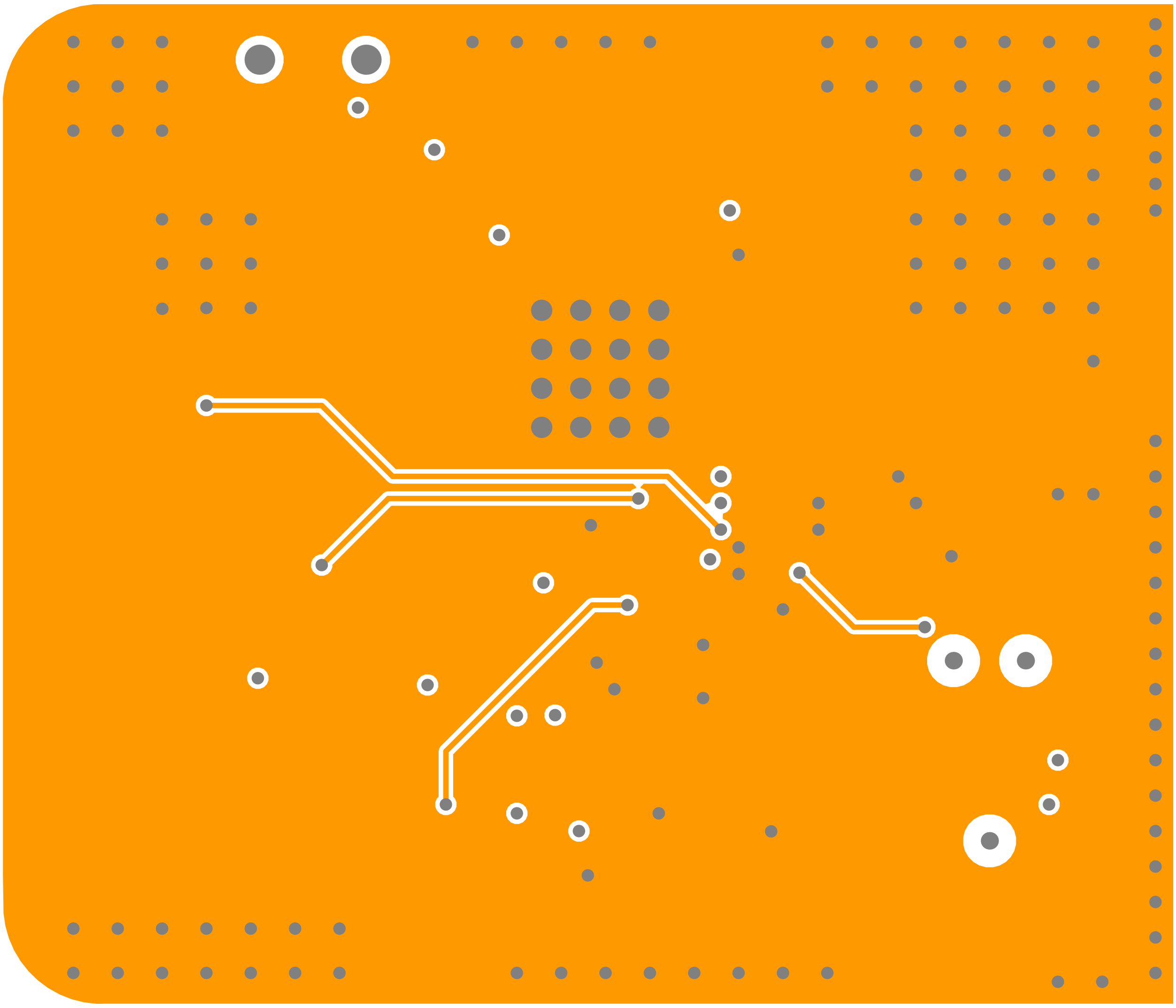


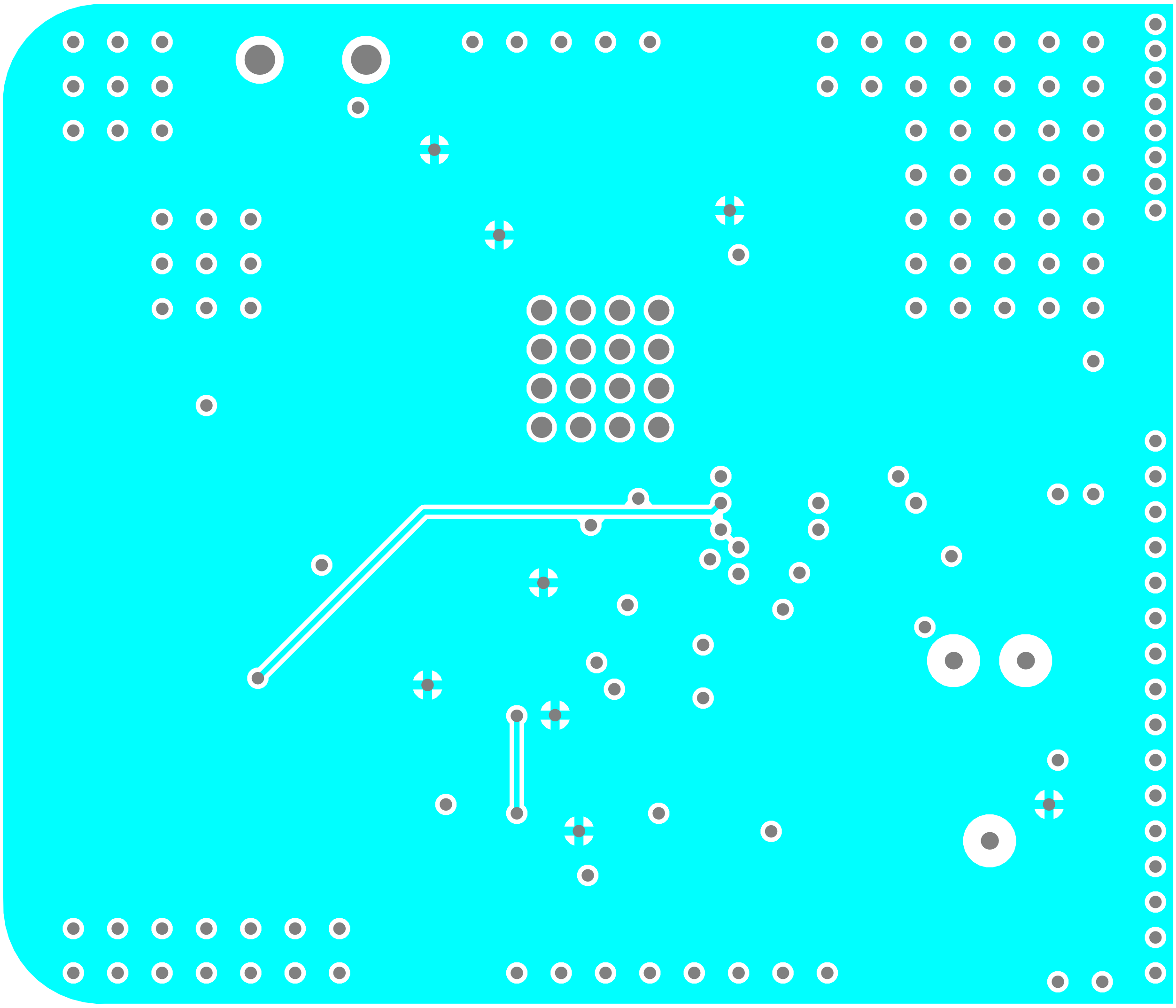


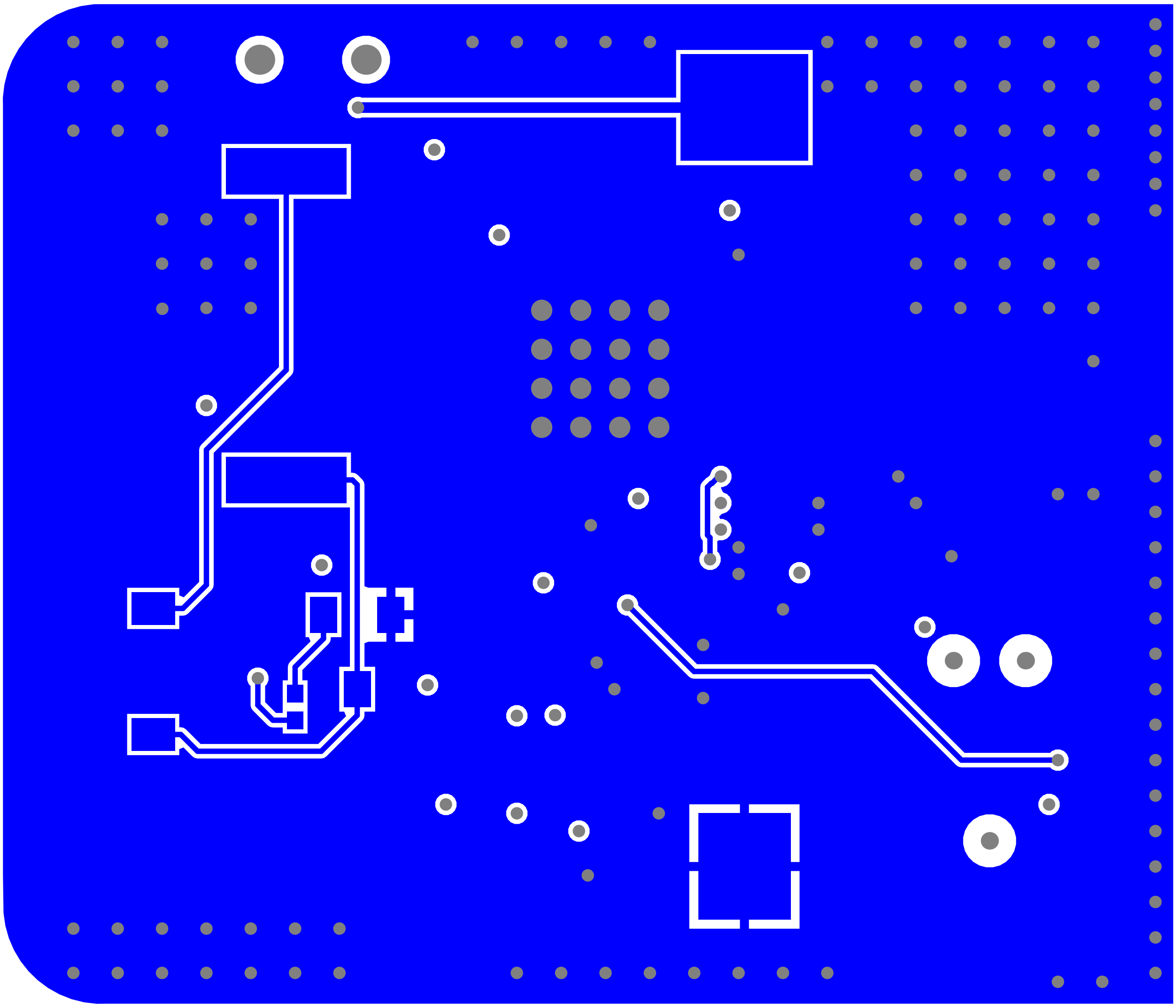

SW4
SW1
SW2
SW3
R5
R4
R1
U1
D3
R2
C16
U2
C11
X2
C8
C7
C5
L2
L1
C1
R7
C17
C3
D2
R6
C4
U4
C2
C14
C6
L3
C15
C12
C10
C9
X1
J2
J1
C13
Ant1


Anabolic Swag
We PR
or ER
+
BT1
R3
Q1
+
D1
SOD-123
LS1









E LTpowerCAD Simulations

LTM4622A Supply Design Summary Report

Vin : 3,6V (min.), 3,7V (nom.), 10V (max.)

Output Rails : Vout1 = 3,325V / 1A (max.) , Vout2 = 1,804V / 1A (max.)

Project Name : LTM4622 MOB Unit

Project Date : 04/2023

Designer : N.R.

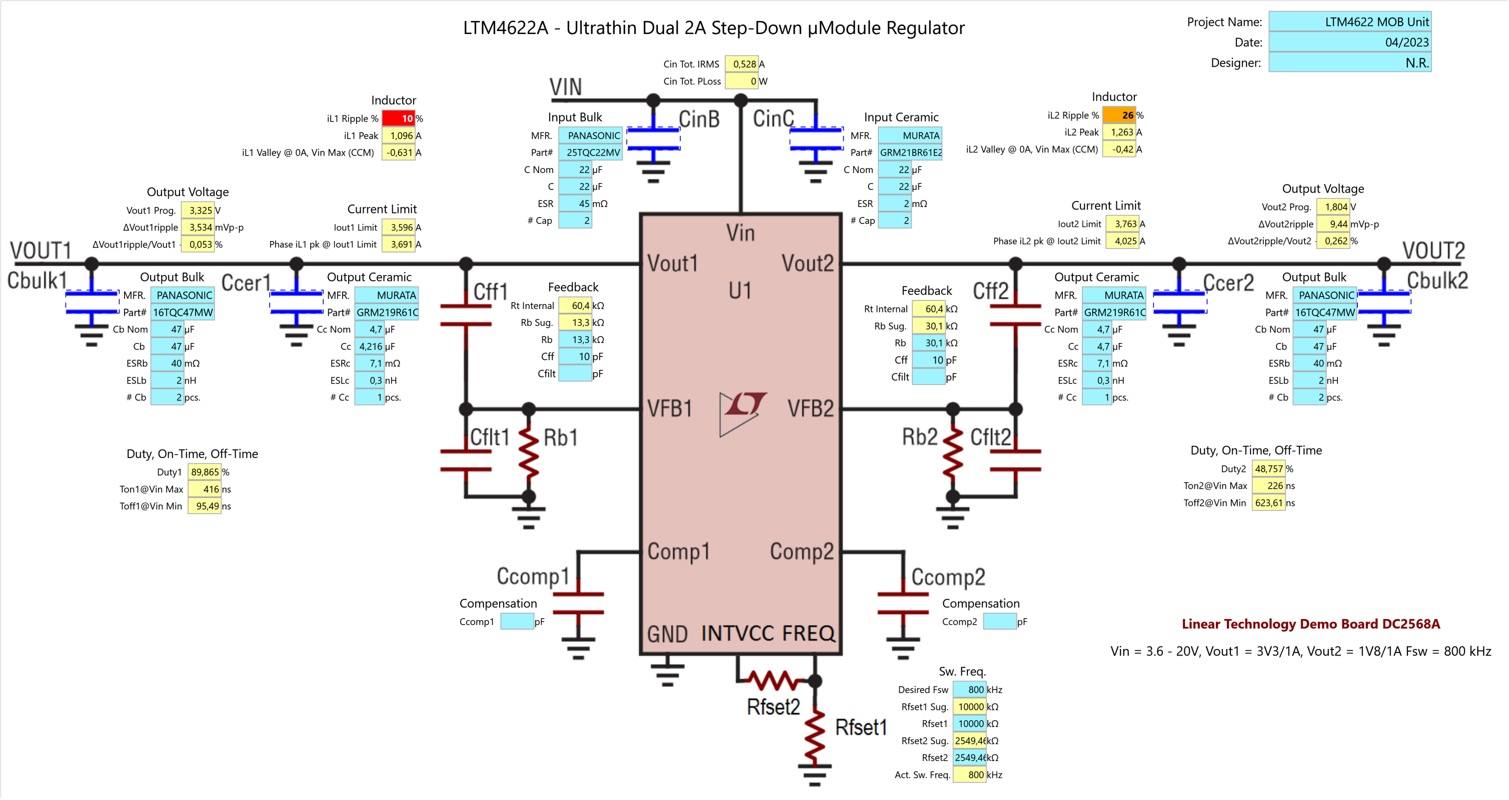
Generated by LTpowerCAD

Printed : 05.15.2023

LTM4622A Solution - Simplified Schematic

Vin : 3,6V (min.), 3,7V (nom.), 10V (max.)

Output Rails : Vout1 = 3,325V / 1A (max.) , Vout2 = 1,804V / 1A (max.)



LTM4622A - Ultrathin Dual 2A Step-Down µModule Regulator

Project Name:	LTM4622 MOB Unit
Date:	04/2023
Designer:	N.R.

Cin Tot. IRMS	0,528	A
Cin Tot. PLoss	0	W

Inductor

iL1 Ripple %	10	%
iL1 Peak	1,096	A
iL1 Valley @ 0A, Vin Max (CCM)	-0,631	A

Input Bulk

MFR.	PANASONIC	
Part#	25TQC22MV	
C Nom	22	µF
C	22	µF
ESR	45	mΩ
# Cap	2	

Input Ceramic

MFR.	MURATA	
Part#	GRM21BR61E2	
C Nom	22	µF
C	22	µF
ESR	2	mΩ
# Cap	2	

Inductor

iL2 Ripple %	26	%
iL2 Peak	1,263	A
iL2 Valley @ 0A, Vin Max (CCM)	-0,42	A

Output Voltage

Vout1 Prog.	3,325	V
ΔVout1ripple	3,534	mVp-p
ΔVout1ripple/Vout1	0,053	%

Current Limit

Iout1 Limit	3,596	A
Phase iL1 pk @ Iout1 Limit	3,691	A

Current Limit

Iout2 Limit	3,763	A
Phase iL2 pk @ Iout2 Limit	4,025	A

Output Voltage

Vout2 Prog.	1,804	V
ΔVout2ripple	9,44	mVp-p
ΔVout2ripple/Vout2	0,262	%

Output Bulk

MFR.	PANASONIC	
Part#	16TQC47MW	
Cb Nom	47	µF
Cb	47	µF
ESRb	40	mΩ
ESLb	2	nH
# Cb	2	pcs.

Output Ceramic

MFR.	MURATA	
Part#	GRM219R61C	
Cc Nom	4,7	µF
Cc	4,216	µF
ESRc	7,1	mΩ
ESLc	0,3	nH
# Cc	1	pcs.

Feedback

Rt Internal	60,4	kΩ
Rb Sug.	13,3	kΩ
Rb	13,3	kΩ
Cff	10	pF
Cflt		pF

Feedback

Rt Internal	60,4	kΩ
Rb Sug.	30,1	kΩ
Rb	30,1	kΩ
Cff	10	pF
Cflt		pF

Output Ceramic

MFR.	MURATA	
Part#	GRM219R61C	
Cc Nom	4,7	µF
Cc	4,7	µF
ESRc	7,1	mΩ
ESLc	0,3	nH
# Cc	1	pcs.

Output Bulk

MFR.	PANASONIC	
Part#	16TQC47MW	
Cb Nom	47	µF
Cb	47	µF
ESRb	40	mΩ
ESLb	2	nH
# Cb	2	pcs.

Duty, On-Time, Off-Time

Duty1	89,865	%
Ton1@Vin Max	416	ns
Toff1@Vin Min	95,49	ns

Duty, On-Time, Off-Time

Duty2	48,757	%
Ton2@Vin Max	226	ns
Toff2@Vin Min	623,61	ns

Compensation

Ccomp1		pF

Compensation

Ccomp2		pF

Linear Technology Demo Board DC2568A

Vin = 3.6 - 20V, Vout1 = 3V3/1A, Vout2 = 1V8/1A Fsw = 800 kHz

Sw. Freq.

Desired Fsw	800	kHz
Rfset1 Sug.	10000	kΩ
Rfset1	10000	kΩ
Rfset2 Sug.	2549,46	kΩ
Rfset2	2549,46	kΩ
Act. Sw. Freq.	800	kHz

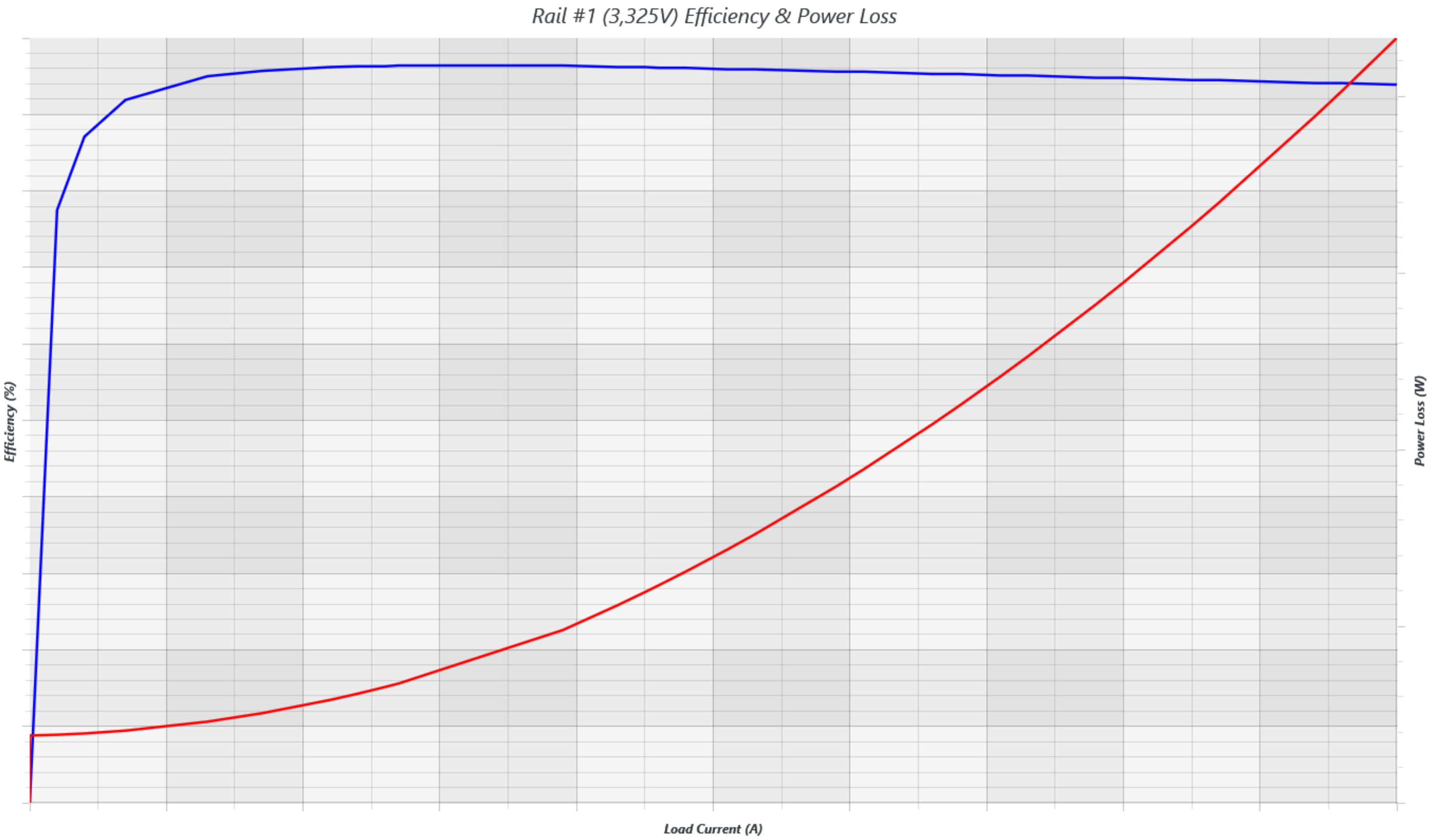
Generated by LTpowerCAD

Printed : 05.15.2023

LTM4622A Solution - Efficiency & Loss Estimations

Rail # 1 : Vin = 5,5V, Vout1 = 3,325V

** Estimations For CCM Mode Only. Inductor AC Losses Entered by User*



Generated by LTpowerCAD

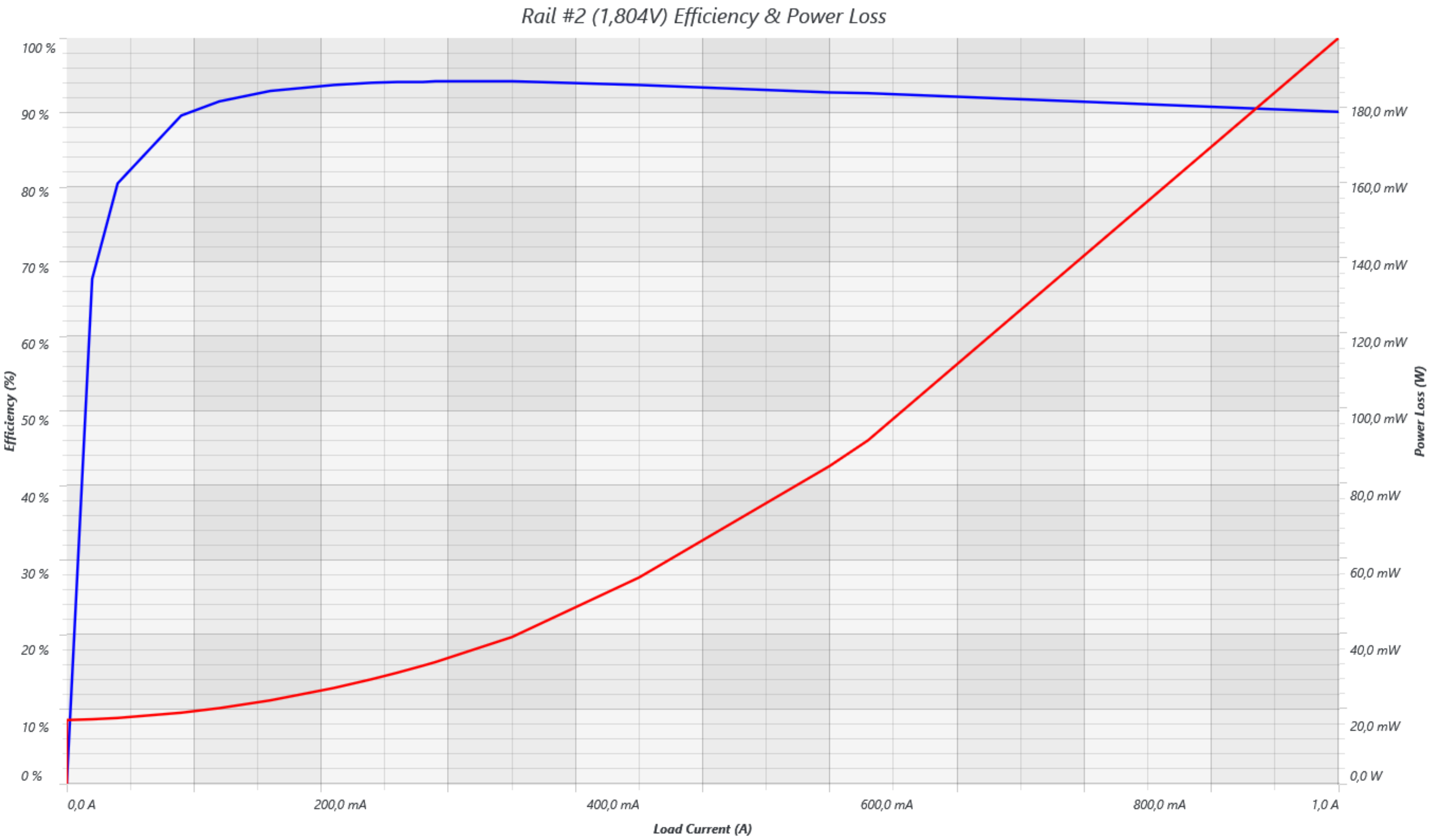
Copyright © 2021 Analog Devices Inc.

Printed : 05.15.2023

LTM4622A Solution - Efficiency & Loss Estimations

Rail # 2 : Vin = 5,5V, Vout2 = 1,804V

** Estimations For CCM Mode Only. Inductor AC Losses Entered by User*



Generated by LTpowerCAD

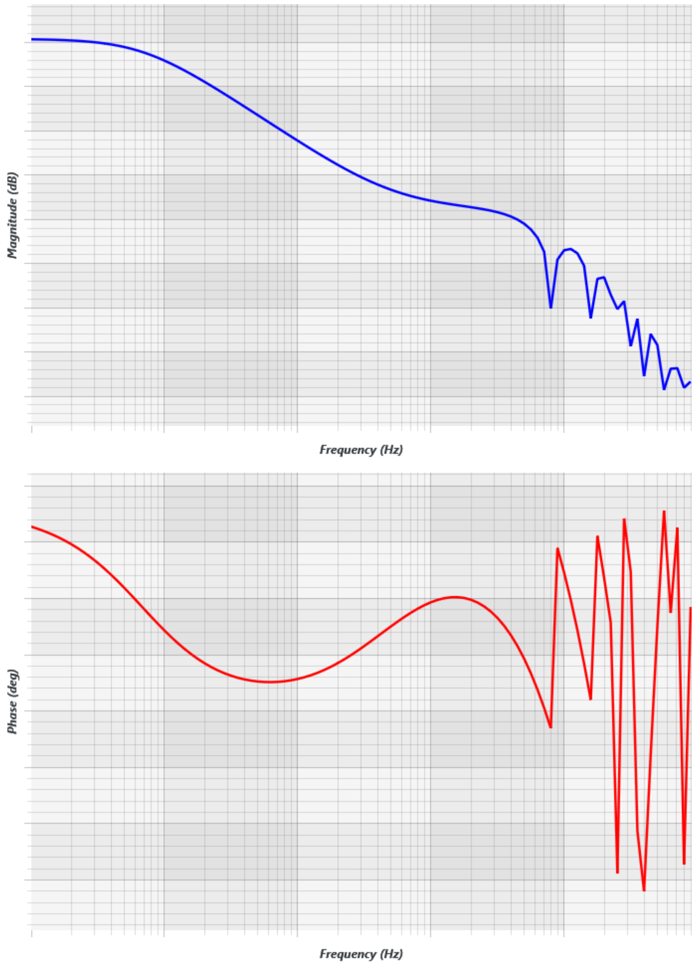
Printed : 05.15.2023

LTM4622A Solution - Loop Gain & Load Transient Estimations

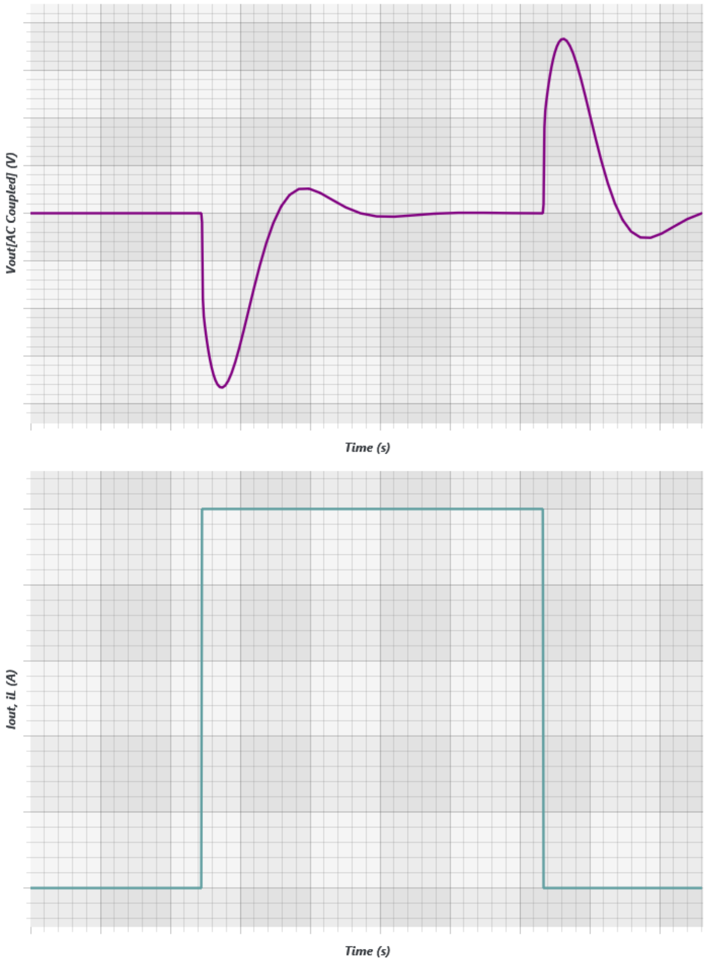
Rail # 1 : Vin = 5,5V, Vout1 = 3,325V, Iout1 = 1A

** Estimations For CCM Mode Only. Estimations Based On Small Signal Avg. Model*

Rail #1 (3,325V) Loop Gain



Rail #1 (3,325V) Load Transient



Generated by LTpowerCAD

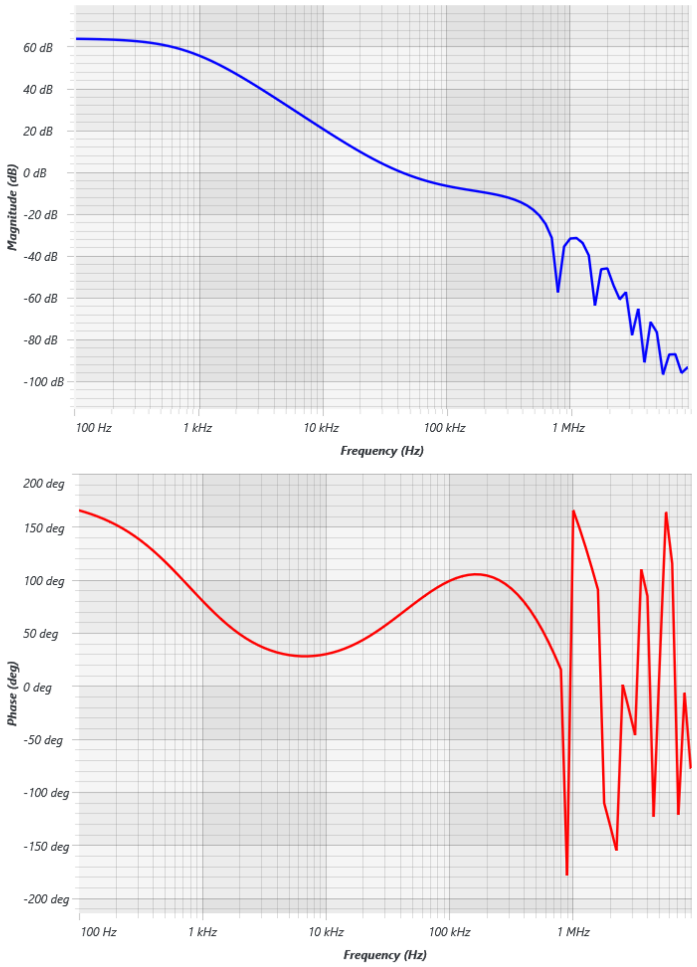
Printed : 05.15.2023

LTM4622A Solution - Loop Gain & Load Transient Estimations

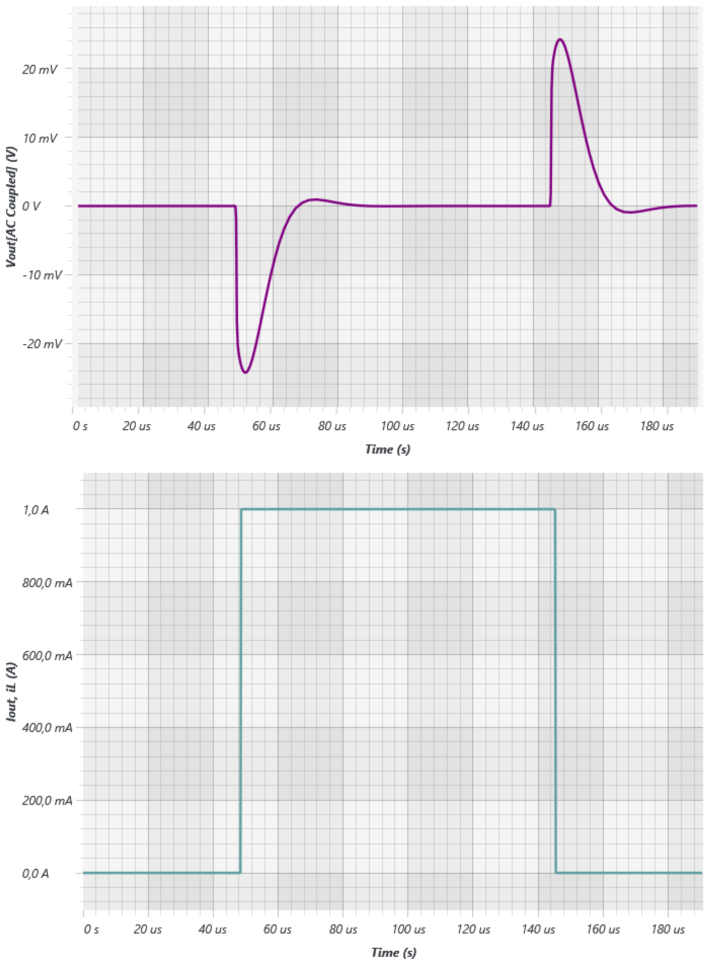
Rail # 2 : Vin = 5,5V, Vout2 = 1,804V, Iout2 = 1A

** Estimations For CCM Mode Only. Estimations Based On Small Signal Avg. Model*

Rail #2 (1,804V) Loop Gain



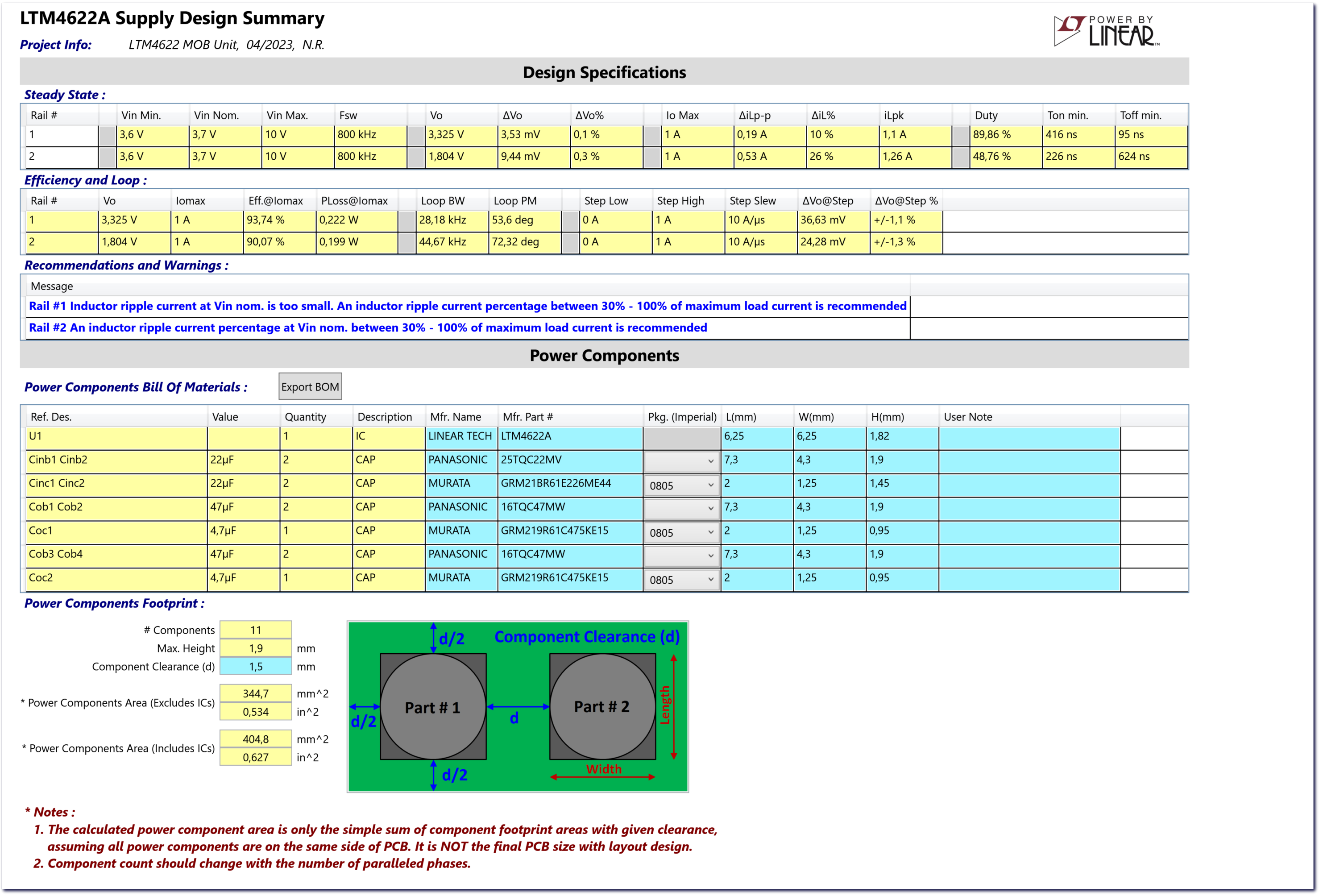
Rail #2 (1,804V) Load Transient



Generated by LTpowerCAD

Printed : 05.15.2023

LTM4622A Solution - Summary



LTM4622A Supply Design Summary

Project Info: *LTM4622 MOB Unit, 04/2023, N.R.*

Design Specifications

Steady State :

Rail #		Vin Min.	Vin Nom.	Vin Max.	Fsw		Vo	ΔVo	ΔVo%		Io Max	ΔiLp-p	ΔiL%	iLpk		Duty	Ton min.	Toff min.
1		3,6 V	3,7 V	10 V	800 kHz		3,325 V	3,53 mV	0,1 %		1 A	0,19 A	10 %	1,1 A		89,86 %	416 ns	95 ns
2		3,6 V	3,7 V	10 V	800 kHz		1,804 V	9,44 mV	0,3 %		1 A	0,53 A	26 %	1,26 A		48,76 %	226 ns	624 ns

Efficiency and Loop :

Rail #	Vo	Iomax	Eff.@Iomax	PLoss@Iomax		Loop BW	Loop PM		Step Low	Step High	Step Slew	ΔVo@Step	ΔVo@Step %
1	3,325 V	1 A	93,74 %	0,222 W		28,18 kHz	53,6 deg		0 A	1 A	10 A/µs	36,63 mV	+/-1,1 %
2	1,804 V	1 A	90,07 %	0,199 W		44,67 kHz	72,32 deg		0 A	1 A	10 A/µs	24,28 mV	+/-1,3 %

Recommendations and Warnings :

Message
Rail #1 Inductor ripple current at Vin nom. is too small. An inductor ripple current percentage between 30% - 100% of maximum load current is recommended
Rail #2 An inductor ripple current percentage at Vin nom. between 30% - 100% of maximum load current is recommended

Power Components

Power Components Bill Of Materials : Export BOM

Ref. Des.	Value	Quantity	Description	Mfr. Name	Mfr. Part #	Pkg. (Imperial)	L(mm)	W(mm)	H(mm)	User Note
U1		1	IC	LINEAR TECH	LTM4622A		6,25	6,25	1,82	
Cinb1 Cinb2	22µF	2	CAP	PANASONIC	25TQC22MV		7,3	4,3	1,9	
Cinc1 Cinc2	22µF	2	CAP	MURATA	GRM21BR61E226ME44	0805	2	1,25	1,45	
Cob1 Cob2	47µF	2	CAP	PANASONIC	16TQC47MW		7,3	4,3	1,9	
Coc1	4,7µF	1	CAP	MURATA	GRM219R61C475KE15	0805	2	1,25	0,95	
Cob3 Cob4	47µF	2	CAP	PANASONIC	16TQC47MW		7,3	4,3	1,9	
Coc2	4,7µF	1	CAP	MURATA	GRM219R61C475KE15	0805	2	1,25	0,95	

Power Components Footprint :

# Components	11	
Max. Height	1,9	mm
Component Clearance (d)	1,5	mm
* Power Components Area (Excludes ICs)	344,7	mm^2
	0,534	in^2
* Power Components Area (Includes ICs)	404,8	mm^2
	0,627	in^2

**** Notes :***

1. ***The calculated power component area is only the simple sum of component footprint areas with given clearance, assuming all power components are on the same side of PCB. It is NOT the final PCB size with layout design.***
2. ***Component count should change with the number of paralleled phases.***

Generated by LTpowerCAD

Printed : 05.15.2023

F Poster

Cellular Asset Tracker

Ultra-low Power Man Over board System

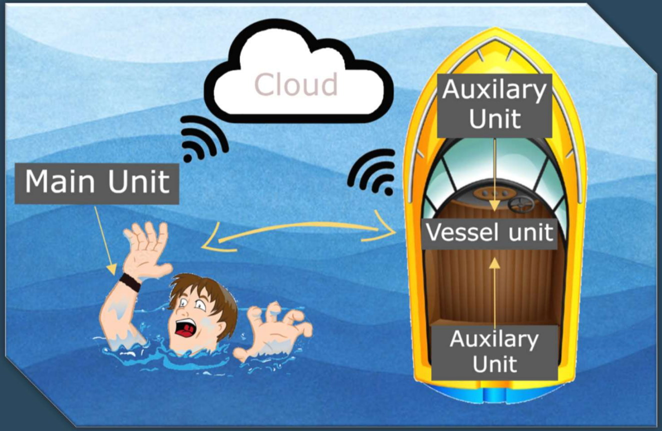
Jonathan H. Tønnesen, Lars Fredrik S. Onsager, Marius H. Olaisen, Njål Rundereim
Department of Electronic Systems

Introduction

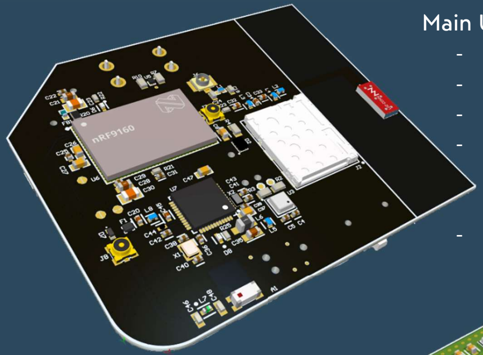
The group has developed a tracking system that automatically detects when users fall into water and sends distress signal to local authorities.

It is possible to expand the system and add on a central unit for the MOB detection system, with the possibility to set up extra units for monitoring critical equipment or give to guests for added personal safety on board.

By leveraging the nRF9160 and nRF52832, these units use Cellular, GPS and Bluetooth communication.

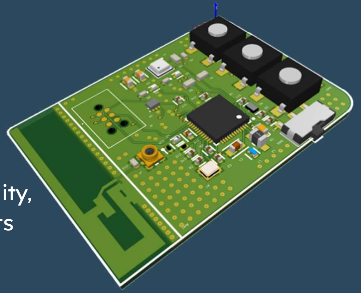


Hardware



Main Unit:

- Cellular connection
- GPS
- Bluetooth
- Temperature, humidity, and pressure sensors
- Analog Devices Micromodule



Auxiliary Unit:

- Bluetooth
- Temperature, humidity, and pressure sensors
- CR2032 Battery

Versatile and Smart Tracking System

- Use as a standalone and reliable tracker.
- Connect to a Vessel Unit for improved coverage.
- Expand the system with Auxiliary Units which can be placed on guests or equipment on board.

Battery Life

Transmission Interval	**Main Unit**	**Auxiliary Unit**
Constant:	7.6 Hrs	34.8 Hrs
4 hr:	159.4 Hrs	254.2 Hrs
8hr:	229.8 Hrs	494.6 Hrs

Note: Calculations are done using a CR2032 battery for auxiliary unit, and a 3.7V 1400mAh battery for the main unit. Draws are gathered from the datasheet. No measured values were used.

Software

Distress signal to a cloud when detecting Man-Over-Board or other emergencies.
Optimized for long battery life with modern battery-saving techniques.
Runs on Zephyr RTOS

Conclusion

This project made considerable progress in developing a Main, Auxiliary and Vessel Unit prototype system. The compact hardware and software design led to the creation of efficient PCBs.
However, due to time restrictions, an entirely functional prototype could not be thoroughly developed and tested.
The firmware will require revisions before integration into the designed prototypes. Although several functions have been set up and tested separately, considerable work remains to make it field-deployable.

The path to the prototype system involved both challenges and successes. The knowledge gained and progress towards a working design confirm the project's potential. The goal going forward is to apply the lessons from this project phase, refine the prototype, and deliver a fully functional, reliable, and efficient system.

Gratitude

The group would like to extend a special thanks to Joakim A. Tønnesen and Kostiantyn Denysenko at Nordic Semiconductor. Their knowledge and expertise helped the group on multiple occasions.

This project has been an incredible journey, and we extend our heartfelt appreciation to all those who guided, supported, and believed in us throughout this transformative academic endeavor.

NTNU

Norwegian University of
Science and Technology

NORDIC
SEMICONDUCTOR