Aalto-yliopisto, Sähkötekniikan korkeakoulu
ELEC-D0301 Protopaja
2023

Loppuraportti / Final Report

Saab Overhead UAV Alert Project
- SKY SENTINEL -

Date: 5.9.2023

Information page

Students
Quoc Quang Ngo
Ville Hirvonen

Project manager
Quoc Quang Ngo

Sponsoring Company
Saab Finland Oy

Starting date
1.6.2023

Submitted date
5.9.2023

Abstract

The need for a comprehensive drone detection solution for military application in urban environments has become more crucial than ever.[1] This project addresses this pressing demand by introducing a cutting-edge drone detection product focused on identifying and mitigating potential drone-based threats, particularly in scenarios like bombings. The paramount objective of this endeavor is to save lives and safeguard critical infrastructure.

Our meticulously designed drone detection product is specifically tailored for military deployment in urban areas, utilizing state-of-the-art technology to ensure swift and accurate identification of drone threats within a range of 15-20 meters. By providing real-time monitoring capabilities, the system offers a proactive defense against drone-based attacks, presenting the user with timely visual, auditory, or tactile alerts the moment a drone presence is detected.

The urgency of this project stems from the escalating use of drones in various malicious activities, leading to devastating consequences in terms of human lives lost and widespread destruction of valuable assets.[2] Addressing this critical issue head-on, our innovative solution aims to minimize the costs of human lives and prevent the catastrophic outcomes of drone-related incidents.

As most drone detection devices are high-end military products tailored for high-budget applications[3], one of the key objectives of this project is to develop a drone detection device from commercially available building blocks that could also have the potential to be mass-manufactured for relatively cheap.

Tiivistelmä

Tarve kattavalle miehittämättömien ilma-alusten (UAV) tai dronejen havaitsemisratkaisulle sotilaskäyttöön kaupunkiympäristöissä on nykypäivänä suurempi kuin koskaan.[1] Tämän projektin tarkoituksena on vastata tähän tarpeeseen esittelemällä huippuluokan teknologialla varustettu UAV-havaitsemislaite, joka keskittyy potentiaalisten ilma-alus- sekä lennokkiuhkien tunnistamiseen ja lieventämiseen, erityisesti pommitusten kaltaisissa skenaarioissa. Tämän projektin ensisijainen tavoite on säästää ihmishenkiä ja olla osana kriittisen infrastruktuurin suojaamista.

Huolellisesti suunniteltu ilma-alusten havaitsemistuotteemme on erityisesti räätälöity sotilaskäyttöön kaupunkialueilla, ja se hyödyntää huipputeknologiaa varmistaakseen nopean ja tarkan lennokkiuhkien tunnistamisen 15-20 metrin etäisyydellä. Tarjoamalla reaaliaikaisia seurantakykyjä järjestelmä tarjoaa ennakointiin pohjautuvan puolustuksen ilma-aluspohjaisia hyökkäyksiä vastaan, esittäen käyttäjälle ajantasaisia visuaalisia, auditiivisia ja taktiileja hälytyksiä heti, kun miehittämättömän ilma-aluksen läsnäolo havaitaan.

Tämän projektin ajankohtaisuus perustuu dronejen käytön lisääntymiseen erilaisissa pahantahtoisissa toiminnoissa, jotka usein johtavat ihmishenkien menetykseen sekä merkittäviin aineellisiin vahinkoihin.[2] Esittämällä potentiaalisen teknologiaratkaisun tähän kriittiseen uhkaan, innovatiivinen ratkaisumme pyrkii ennaltaehkäisemään miehittämättömien ilma-alusten ja dronejen aiheuttamia vahinkoja niin ihmishengille kuin myös aineelliselle omaisuudelle.

Koska useimmat ilma-alusten havaitsemislaitteet ovat korkean budjetin sotilaallisiin sovelluksiin räätälöityjä usein hyvinkin monimutkaisia laitteita[3], yksi tämän projektin keskeisistä tavoitteista onkin kehittää kyseinen havaitsemislaite kaupallisesti saatavilla olevista komponenteista sekä edullisemmista teknologiaratkaisuista, mikä tarjoaa myös mahdollisuuden laitteen potentiaaliselle massatuotannolle ja kaupallistamiselle tulevaisuudessa.

Table of Contents

1. Introduction **5**
2. Objective **6**
3. Hardware Development **7**
3.1. Component Selection 7
3.2. Circuit Design 8
3.3. PCB Layout 13
3.4. Hardware Testing 14
4. Firmware Development **15**
4.1. Code Overview & Explanation 15
4.2. Developing Process 17
4.3. Code Unit Testing 19
5. Enclosure Design and 3D Modelling **22**

6. User Experience 25
7. Reflection of The Project 26
7.1. Reaching Objective 26
7.2. Timetable 27
7.3. Risk Analysis 27
7.4. Challenges & Troubleshooting 28
7.4.1. Firmware Challenges 28
7.4.2. Hardware Challenges 29
7.4.3. Testing Phase Challenges 29
8. Discussion and Conclusions 29
8.1. Key Insights & Lessons Learned 30
8.2. Project Conclusions 30
8.3. Acknowledgments 32
9. List of Appendices 32
10. References 32

1. Introduction

Unmanned aerial vehicles (UAVs), more commonly known as drones, have rapidly emerged as a transformative technology, revolutionizing various industries and applications worldwide. As these versatile machines become increasingly prevalent, so do the concerns surrounding their potential misuse for nefarious activities. This project sets out to address a pressing need in contemporary security efforts—the development of an innovative alerting system for detecting UAVs overhead, aiming to safeguard public safety and critical infrastructure.

1. **Background and Significance:** With the exponential growth in drone usage across diverse sectors, the need for a reliable UAV detection system has become paramount. Drones offer remarkable advantages in areas such as surveillance, logistics, and disaster response, but they also present security challenges when exploited by malicious actors. Incidents of unauthorized drone flights near sensitive locations and the potential for drone-based threats, such as bombings and reconnaissance missions, have raised significant concerns among security agencies and the public alike.[2]

2. **Historical Context:** The history of unmanned aerial vehicles dates back to the early 20th century, but it was only in recent decades that advancements in technology led to their widespread adoption. Initially deployed primarily for military reconnaissance and surveillance purposes, drones have since found applications in various industries, including agriculture, filmmaking, and e-commerce delivery services.[4] Unfortunately, the increase in drone availability has also contributed to security breaches, privacy infringements, and unauthorized use cases.

3. **Technological Advancements:** The rapid progress in drone technology has made UAVs more accessible, affordable, and capable than ever before. Modern drones boast extended flight times, impressive payload capacities, and advanced communication systems. In parallel, counter-drone technology has evolved to detect and mitigate potential threats posed by drones. Our project contributes to this technological landscape by developing an alerting system that leverages cutting-edge methods to accurately detect UAVs and provide timely alerts to users.

4. **Contemporary Usage and Challenges:** The widespread adoption of drones across various sectors brings both opportunities and challenges. While drones have streamlined tasks and improved efficiency, their misuse has raised concerns regarding privacy violations, airspace security, and public safety.[2] Incidents of drones trespassing restricted areas, interfering with commercial flights, and carrying illicit payloads underline the necessity of effective detection and mitigation measures.

Our project focuses on designing an alerting system that not only identifies UAVs overhead but also ensures portability, enabling users to carry it conveniently from one location to another. By offering a portable solution, we aim to enhance the adaptability and usability of our detection system, making it an indispensable tool for security personnel, law enforcement, and critical infrastructure protection.

2. Objective

The objectives for the capabilities of the device are:

- Detecting 2.4 GHz Drones (RF Method): Making sure the device can spot drones using radio frequency (RF) signals.

- Minimum 20m Detection Range: Ensuring the device can pick up drone signals from at least 20 meters away.
- High Portability: Making the device easy to carry around, so it's not a hassle to use.
- Versatile Alerting Options: Giving the device different ways to make alerts, so people can be warned effectively.
- Low Production Costs: Keeping the cost of making the device low.
- Suitable for Military Use: Making sure the device is good to go for military applications, especially in battlefields.

The targets for the project are well discussed with the company in the beginning of the project. However, as the project advanced, certain aspects of the objectives were modified in alignment with our operational capacities and project timeline. These adaptations were made to ensure feasibility and project progression.

3. Hardware Development

3.1. Component Selection

As this device is aimed to be built upon commercial building blocks, the hardware and component selection is largely based on the relative price and widespread availability for each hardware part. The key hardware components the device currently utilizes are:

- BMD-301 Bluetooth module: Based on the nRF52832-chip, this module allows for easy-to-use Bluetooth low-energy technology, customizable general purpose hardware control as well as an externally connected antenna.
- External antenna: The device is aimed to function with an external antenna connected to the BMD-301 module, allowing for a high level of customizability. In this case, we opted for an omnidirectional dipole antenna by Taoglas Limited. The antenna is connected to the BMD-301 via a U.FL-SMA connector.
- CR2477T coin cell battery: The motivation behind this type of power source comes

from the widespread availability and reliability of the CR2477 -line of coin cell batteries, making it effortless to swap out and change the device's batteries when needed. The need for specifically a battery-powered solution is based on the desired easy portability and small size of the device.

- Tag-Connect port: Utilizing a port for the Tag-Connect -line of programming cables allows for easy and non-intrusive custom programming of the BMD-301 module.

- Various alerting peripherals: The device utilizes various human-sensory based alerting methods, such as an LED, a buzzer and a vibration motor. These components are chosen mainly based on their commercial availability and small size and can be swapped out for similar alternative components if needed.

- Various general circuit components: The circuit inside the device also requires various general components, such as capacitors, resistors and MOSFETs. Most of these are chosen to be of 0603-size, as this is a widely available, affordable and small enough component size for this use-case.

Potential useful add-ons or modifications that could be made to the components and hardware include a built-in antenna in the PCB directly in addition to or completely replacing the external antenna (or alternatively, a different module with a built-in antenna) as well as a different, possibly more powerful LED with the possibility of automatic brightness adjustment based on environment (by adding a brightness sensor peripheral into the PCB) or night-vision mode by using an infrared LED. These add-ons, however, have not been implemented in this version of the project.

3.2. Circuit Design

The high-level idea behind the device's hardware is quite simple and is illustrated in the block diagram below.

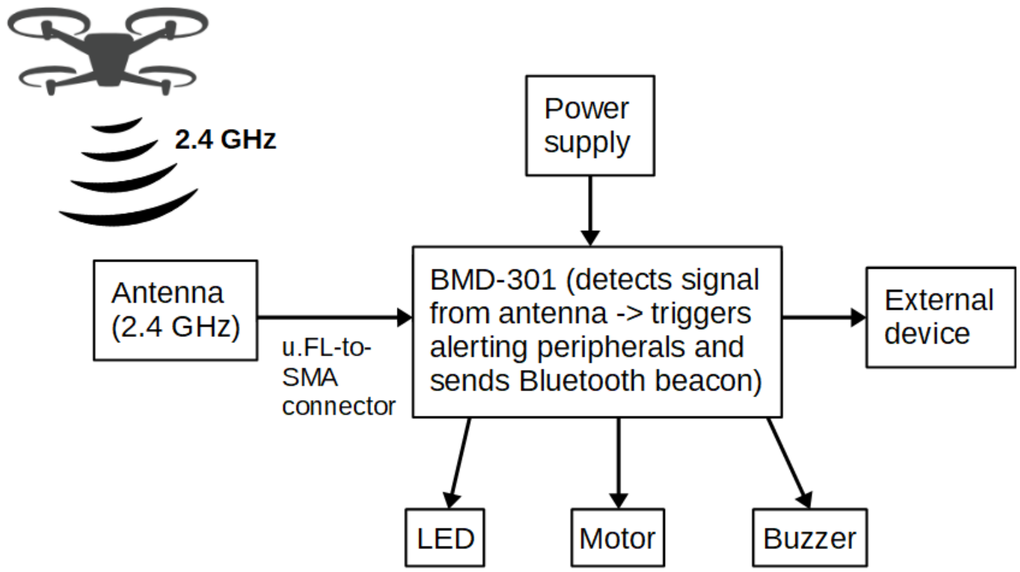

Figure 1: Block diagram illustrating the high-level concept behind the device.

The circuitry for implementation of the above block diagram can be seen as a whole in the schematic below. All of the electronic design is done using the KiCad-software, but can easily be adapted for other programs as well. The components are chosen based on the section "3.1. Component Selection".

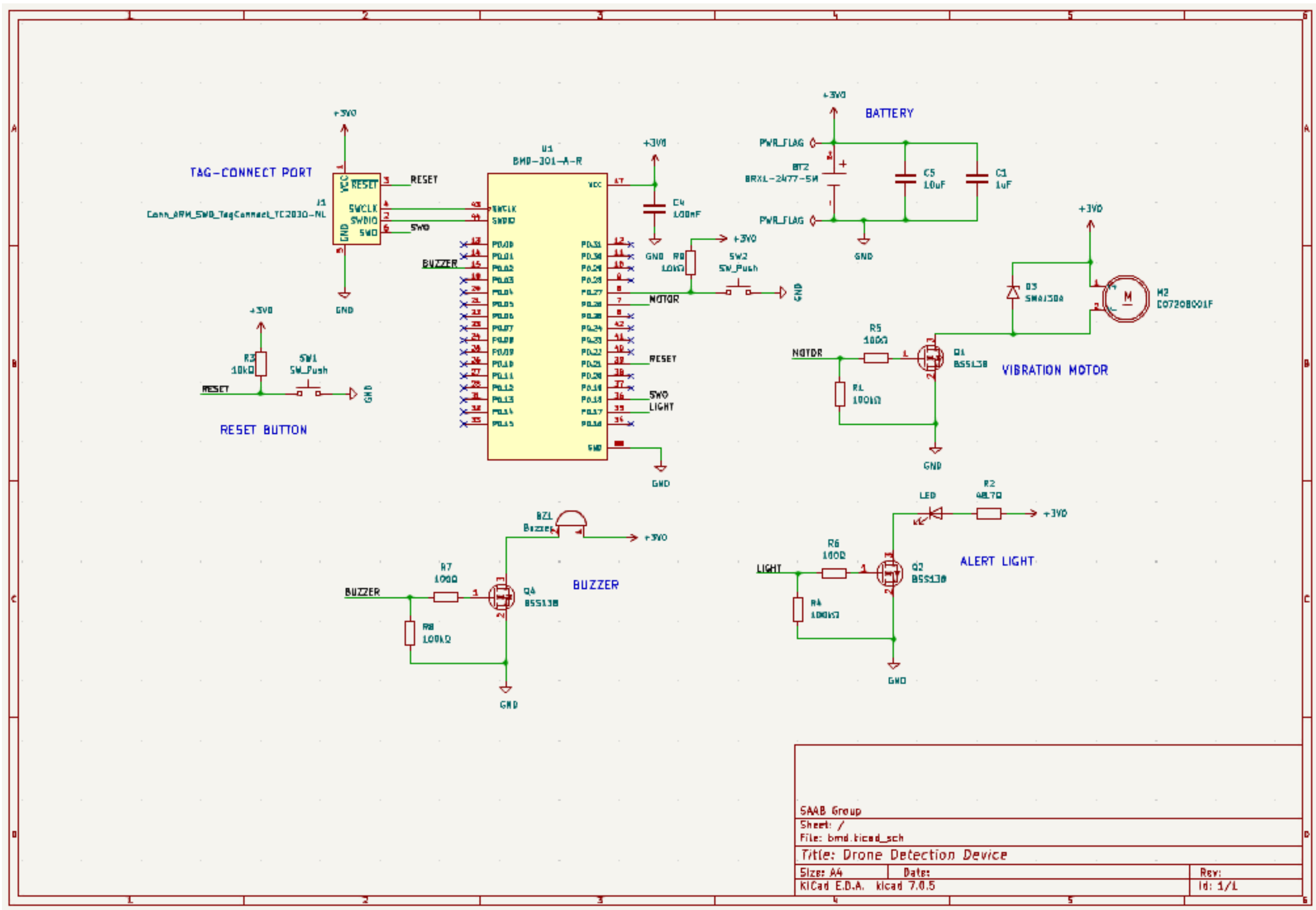

Figure 2: Schematic of the entire device's circuitry.

First, we present the circuitry for the battery. The intended power supply for the device is a 3 V CR2477 -coin cell battery and is kept in place by a metallic BRX1-2477-SM battery holder. Decoupling capacitors of various capacitances are placed in parallel with the power supply to smooth out voltage fluctuations and reduce high frequency noise in the power supply signal.

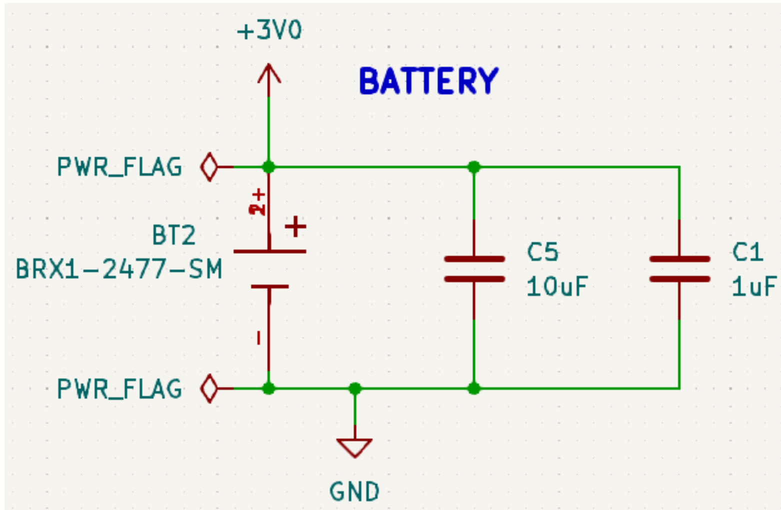

Figure 3: Schematic of power supply circuitry.

Next, we have the BMD-301 module and its programming circuitry, which is responsible for the most important functionalities of the device. The triggering of the alerting peripherals is done through the general purpose input/output (GPIO) pins of the BMD-301, in particular, pins P0.02, P0.17 and P0.26. Connected to P0.27, there is a button that can be custom-programmed for whatever purpose necessary. The Tag-Connect programming port is connected to the module via its serial wire debug interface. A reset button for debugging purposes is connected to P0.21 and triggers a hardware reset by pulling P0.21 low when the button is pressed.

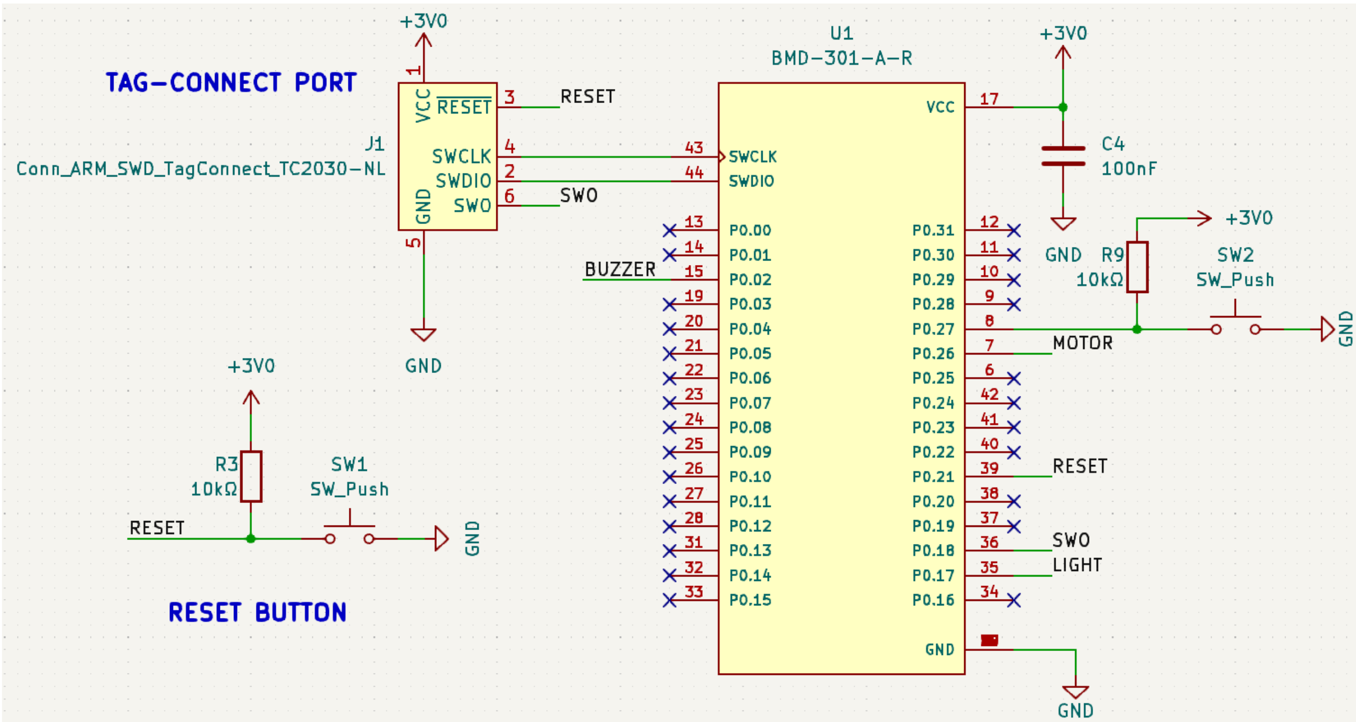

Figure 4: Schematic of BMD-301 module and programming circuitry.

For the alerting peripherals - LED, vibration motor and buzzer - the logic behind the circuitry for each is the same. When the associated pin is set to high, the MOSFET (BSS138) activates and current is allowed to flow from the power supply to ground through the pins of the peripheral device, which activates it. Resistors are placed in the circuitry for each peripheral for optimal functionality. A TVS diode (SMAJ30A) is placed in parallel with the motor to protect the module from any high voltage spikes caused by the motor being turned on/off.

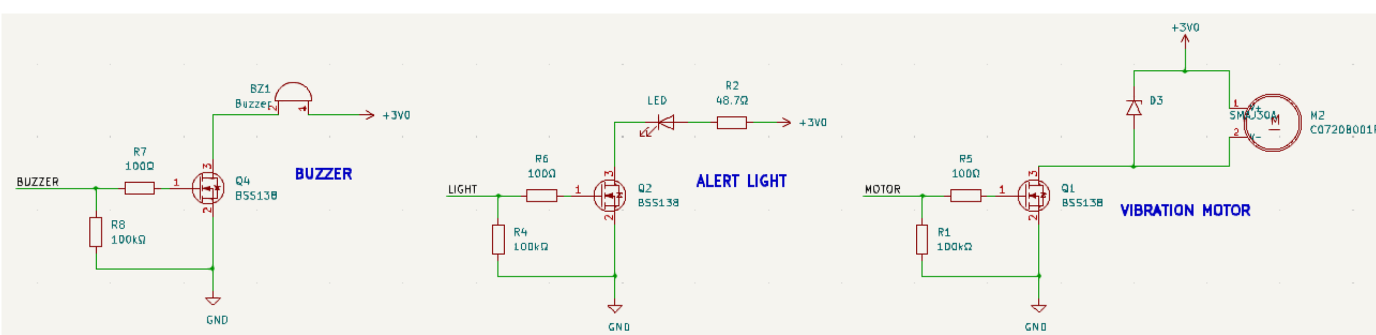

Figure 5: Schematic of the alerting peripheral devices.

3.3. PCB Layout

The next step in the hardware development process was designing the PCB layout (also using KiCad) and assembling the components to the board manually. An image of the PCB layout design can be seen below.

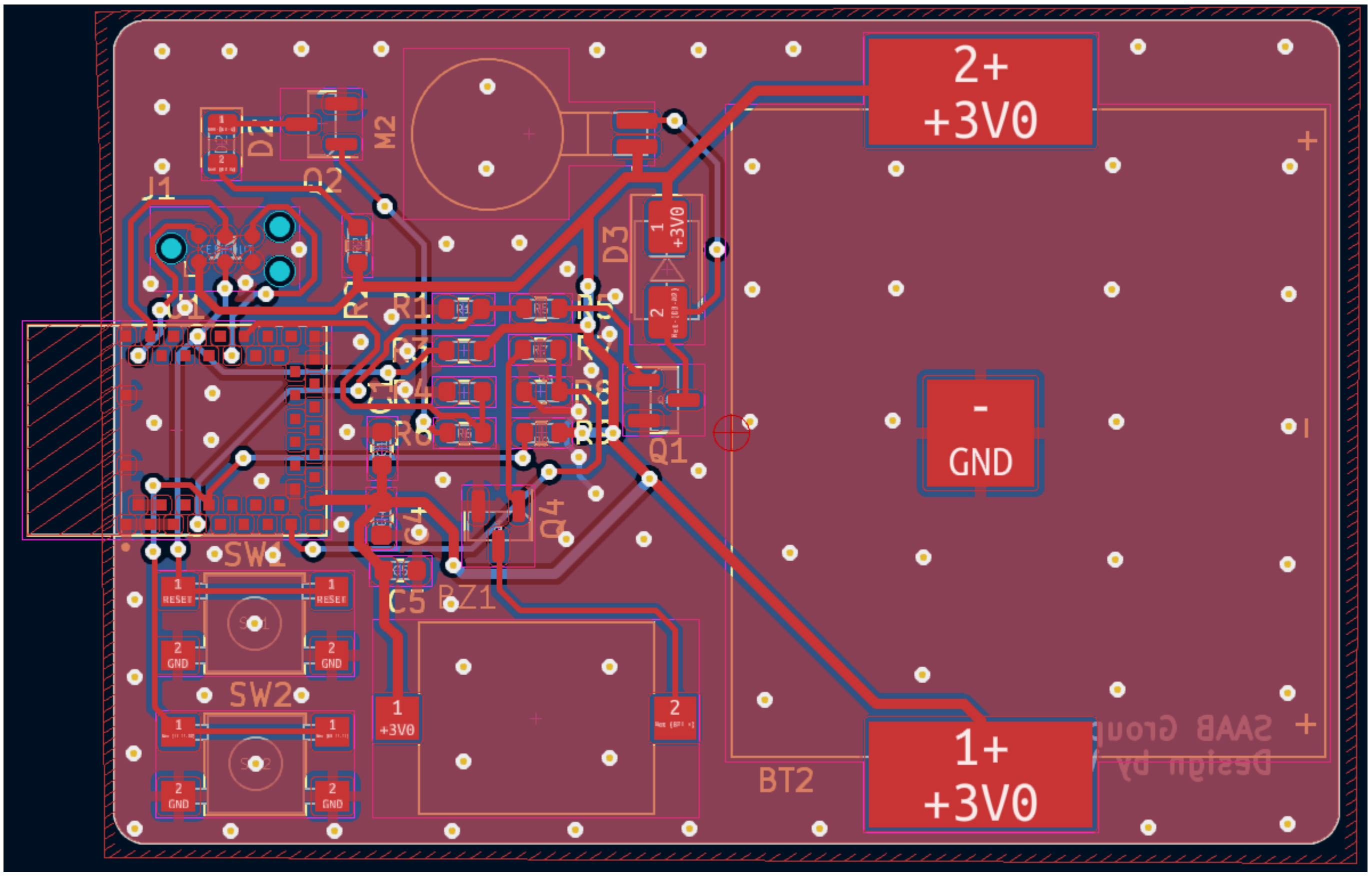

Figure 6: The device's PCB layout.

The main goal with this PCB layout is compactness and simplicity. All components are placed as close to possible as realistically possible, while still keeping in mind the actual assembly process. The red in the above layout is the ground plane that connects to all ground pins of each component. Vias are placed all around for unobstructed ground access.

Although not strictly necessary for this application due to its external antenna connector, the BMD-301 module (on the left above) is placed according to the RF requirements of the similar BMD-300 module, which has an internal antenna. This is simply because it allows the BMD-301 to be replaced with the BMD-300 (or a similar module with an internal antenna) in a later revision of the device if the need for an external antenna is not necessary anymore, without having to alter the PCB layout in any way.

After the required manufacturing files had been sent to JLCPCB for production and the order arrived, the next step was the assembly of all the electronic components to the circuit board. This was done using a stencil that came with our order, appropriate solderpaste and a reflow oven. After a few attempts at the assembly process, we eventually had a working circuit board as shown below.

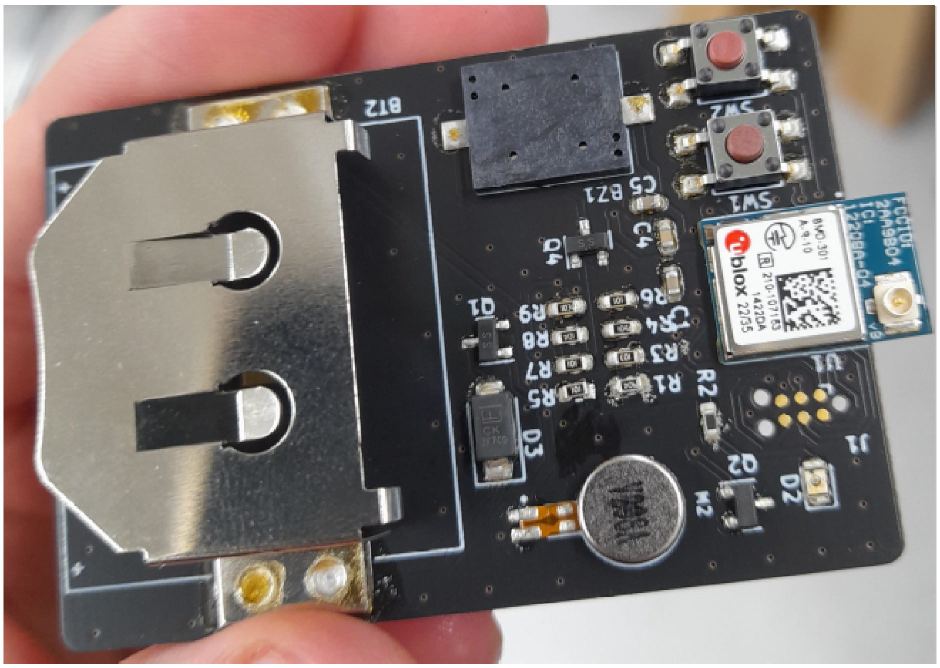

Figure 7: PCB with all the components assembled.

3.4. Hardware Testing

The last step in hardware development was applying various tests to find out which parts of the device work as intended and which do not. After doing some multimeter testing, the diode and LED were wound to having been placed the wrong way due to misread notation. After desoldering and placing them in the correct way, the board was then tested by uploading some simple test code to it.

Testing was also implemented for the antenna, mainly to see what the functional range of the device would be. This was done by our team in the woods near Aalto University, where we hoped there wouldn't be many other signals. The main test we did was use a drone controller operating at 2.4 GHz frequency, and have the device detect the signal at various ranges.

After some time, we were able to still detect a clear signal at nearly 100 meters.

4. Firmware Development

4.1. Code Overview & Explanation

The developed firmware of product is built on top of the ble_app_uart example template from nRF5 SDK[5]. The source code also take advantage of the Beacon Transmitter Sample Application[6], and RSSI Viewer app for nRF Connect for Desktop source code[7].

The developed firmware for our product introduces several significant features and functionalities that distinguish it from the original example template:

- RSSI Signal Scanning: The firmware incorporates robust RSSI signal scanning capabilities, enabling the device to detect and assess the strength of nearby wireless signals. This functionality forms the foundation for our drone detection mechanism.

- Drone Detection: Building upon the RSSI scanning capability, the firmware includes a simplified 2.4GHz drone detection module. Through signal analysis and pattern recognition, the device can identify the presence of nearby drones and distinguish them from other wireless sources.

- Beaconing: This functionality allows the device to transmit periodic signals, which can be utilized for location tracking, proximity sensing, and other context-aware applications.

- GPIO and PWM Control: To enhance the user experience, the firmware provides GPIO and PWM control features. These capabilities enable the device to trigger alerting mechanisms, such as lights or alarms, when a drone is detected. Users can customize these alerts based on their preferences.

- Wireless DFU (Device Firmware Update): The firmware includes support for wireless Device Firmware Updates. This feature streamlines the process of updating the device's firmware over-the-air, enhancing convenience and enabling future improvements without requiring physical connections.

These combined features empower our product to serve as an advanced wireless detection and alert system. By intelligently detecting drones, transmitting signals, and offering customizable alerting options, our firmware adds substantial value to the original example template. This aligns with our project's objectives to create a comprehensive solution for drone detection and interaction.
We adopted a modular approach to develop and validate individual functionalities of the product. Subsequently, these separate functionalities were integrated into a unified codebase. In the subsequent section, we will present the results of our code unit testing.

In terms of the code, the primary function responsible for event handling is the uart_loopback() function. Below is presented a high-level overview of how this function is used in this device.

The code for this begins with variable initiation and measuring all channels for their RSSI values. It then stores the data in a buffer variable as shown below.

```
void uart_loopback()
{
        uint8_t sample;
        uint32_t value;
        value = 50;
        uint8_t count1 = 0 ;

                uint8_t buf[81], maxvalue=0, minvalue=127, diffvalue;
                for (uint8_t i = min_channel; i <= max_channel; ++i)
                {
                        sample = rssi_measurer_scan_channel_repeat(i);
                        buf[i]=sample;
                        maxvalue=(sample>maxvalue)?sample:maxvalue;
                        minvalue=(sample<minvalue)?sample:minvalue;
                }
```

Figure 8: Variable initiation and RSSI scanning.

Next, each channel's RSSI value is assigned an appropriate symbol that represents its approximate value.

```
                char *symbols[5]={" ", ".", ":", "|", "X"};
                for (uint8_t i = min_channel; i <= max_channel; ++i)
                {
                  uint8_t choice=(4*((uint32_t)buf[i]-(uint32_t)minvalue))/(uint32_t)diffvalue;
                  uart_puts(symbols[choice]);
```

Figure 9: Designation of symbols for representing the RSSI value of each channel.

The following lines of code shown below are then used to catch any RSSI peaks. Note that the lower the value, the stronger the signal as the scanning value is negative.

```
if (buf[i]<50)
{
  ++count1;
}
```

Figure 10: Catching peaks in RSSI values.

The next lines in the code are for handling the alerting event, which takes place when 10 peaks occur in all the channels. As the radio for RSSI scanning cannot be turned on when beaconing, we must first turn off the radio, after which the beacon can be sent out. Along with this, a PWM control signal for the LED and buzzer as well as turning on the motor with GPIO control are also initiated. After a brief delay, we then turn off everything before turning on the radio again to continue signal scanning.

```
if (count1>10)
{

    reset_rssi_measurer_configure_radio();
    nrf_sdh_disable_request();
    nrf_sdh_disable_request();
    ble_stack_init();
    advertising_init();
    advertising_start();
    nrf_gpio_pin_set(motor_pin);
    nrf_delay_ms(sweep_delay);
    app_pwm_enable(&PWM1);
    APP_ERROR_CHECK(app_pwm_channel_duty_set(&PWM1, 1, value));
    nrf_delay_ms(sweep_delay);
    nrf_gpio_pin_set(led_pin);
    NRF_LOG_INFO("Drone detected.");
    nrf_delay_ms(1000);
    nrf_sdh_disable_request();
    nrf_sdh_disable_request();
    rssi_measurer_configure_radio();
    app_pwm_disable(&PWM1);
}else{

  nrf_gpio_pin_clear(led_pin);
  nrf_gpio_pin_clear(motor_pin);
}
```

Figure 11: Handling of the alerting event, which occurs when 10 RSSI peaks are detected.

4.2. Developing Process

The developing process of the product's firmware go through these steps:

1. Get used to the development kit:

The initial step involves becoming acquainted with the development kit's hardware and software components. This includes understanding the capabilities and functionalities of the underlying platform.

2. Developing RSSI signal scanning:

   Building upon the foundation of the RSSI Viewer app for nRF Connect for Desktop. The firmware development begins with the creation of RSSI signal scanning functionality. This step establishes the core capability for detecting and assessing nearby wireless signals.

3. Developing beaconing:

   The next stage involves the integration of beaconing functionality. This enables the device to transmit periodic signals, enhancing its potential for context-aware applications and interactions.

4. Developing GPIO control:

   The firmware extends its capabilities by adding GPIO control features. This allows the device to activate the motor for alerting.

5. Developing PWM control:

   Expanding further, the firmware incorporates PWM control capabilities. In order to activate the buzzer, PWM controlling functionality is needed.

6. Developing detection algorithm:

   The algorithm of the product is simple as we assume that in the battlefield environment, there is less or no 2.4ghz signal. Therefore, the detecting of drone is done through scanning and listening for RSSI peaks.

7. Developing Wireless DFU:

   The firmware development process includes the incorporation of Wireless Device Firmware Update (DFU) capabilities. This feature streamlines future firmware updates through wireless communication channels.

8. Uploading and debugging:

As the final stage of the development process, the firmware is uploaded to the developed hardware. Rigorous testing and debugging procedures are carried out to ensure optimal functionality and performance.

4.3. Code Unit Testing

1. RSSI Signal Scanning testing output:

   The below image showing the RSSI scanning on 80 frequency channels within the 2.4Ghz frequency range. After each scan, the terminal will print out the approximation value of the channels represented by 5 symbols ‘ ’, ‘.’, ‘:’, ‘|’,’X’ from low to high value. Each column represents each channel, and each row represents each scan. Emil Fihlman, our course's teaching assistant, is the credited creator of this terminal display designed for debugging and testing.

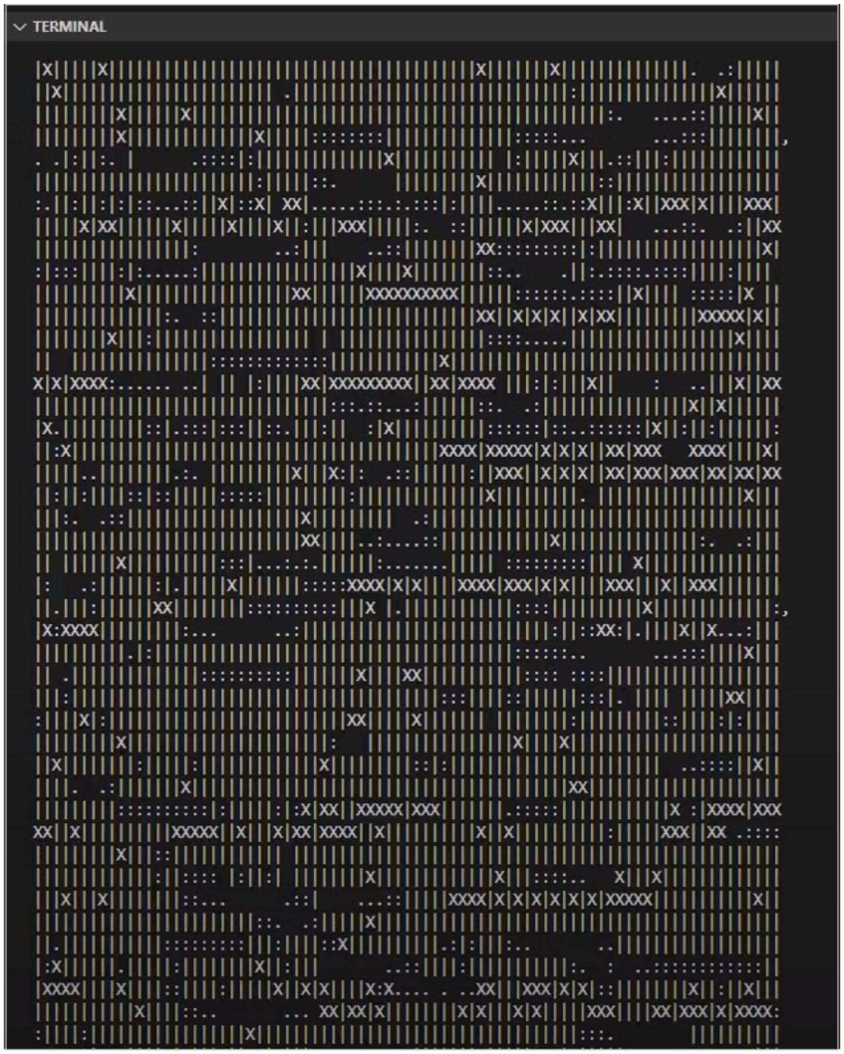

Figure 12: Output terminal for testing code.

2. Beaconing testing output:

   The below image shows the screenshot of the RSSI Connect app on an android phone that succeed to catch the beacon named "DroneDetected" that we send out with the code.

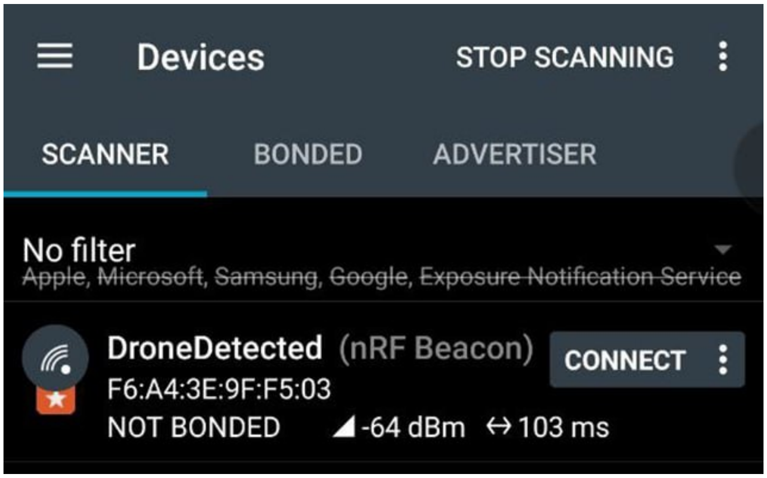

Figure 13: nRF Connect app on Android screenshot showing the alerting beacon.

3. Device Firmware Upgrade

The images below show the beacon used for DFU named “DfuTarg” and the success of uploading new firmware to the product wirelessly

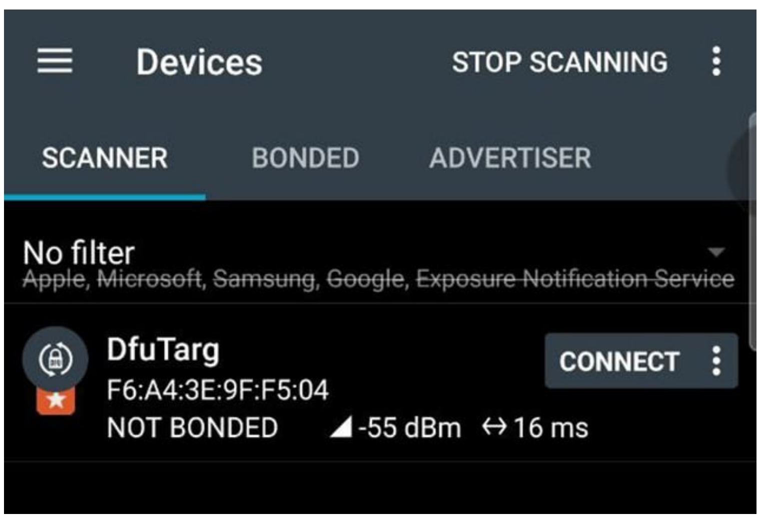

Figure 14: nRF Connect app on Android screenshot showing the beacon for DFU.

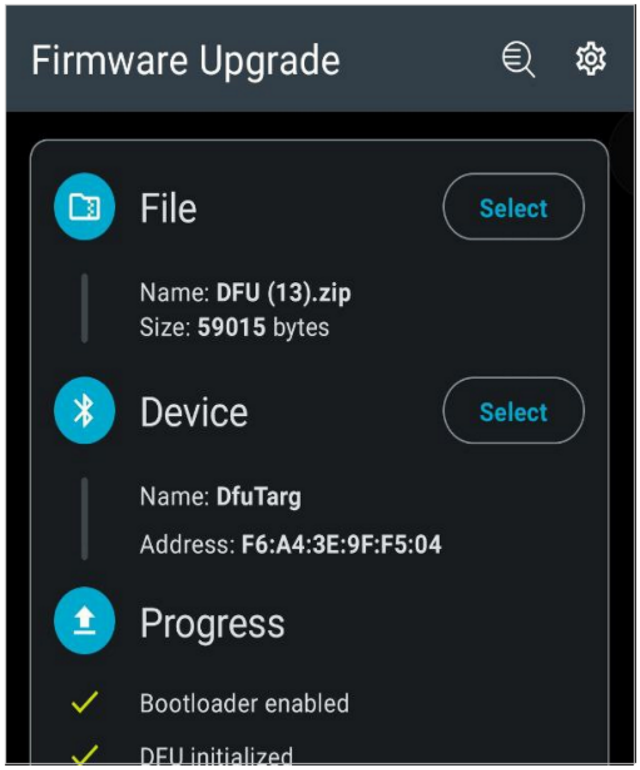

Figure 15: nRF DFU app on Android screenshot showing the success of wireless code upgrade.

5. Enclosure Design and 3D Modelling

The last step in our design phase was designing the enclosure for the device. The main principles we wanted to go by regarding this were simplicity and practical functionality. It was important that the circuit board could easily be taken outside the device, for example, in order to replace a battery or debug something through the Tag-Connect port. Because of this, the enclosure design consists of slide rails that allow the board to be easily slid in and out of the enclosure, while still keeping it firmly in place. Slide rails are also implemented for the side wall and lid.

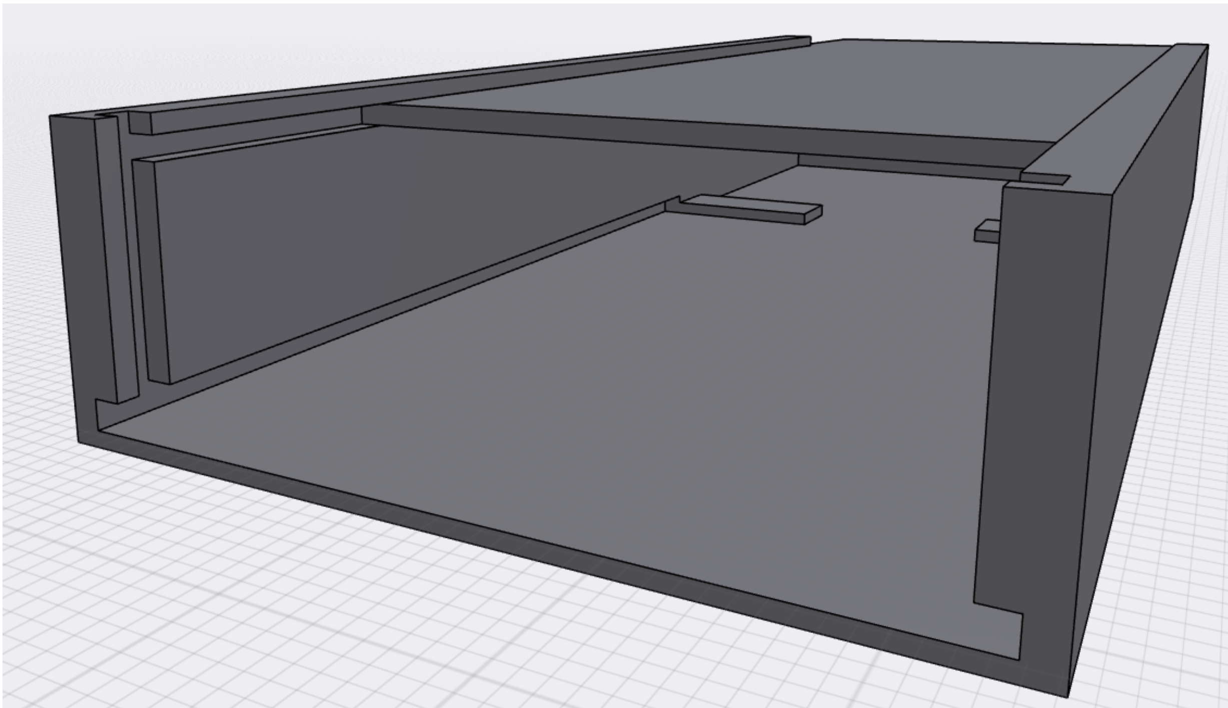

Figure 16: Slide rails for the enclosure.

For connecting our external antenna, we added a hole for the antenna connector. This also allows the entire u.FL-to-SMA connector cable to be placed inside the device and kept protected.

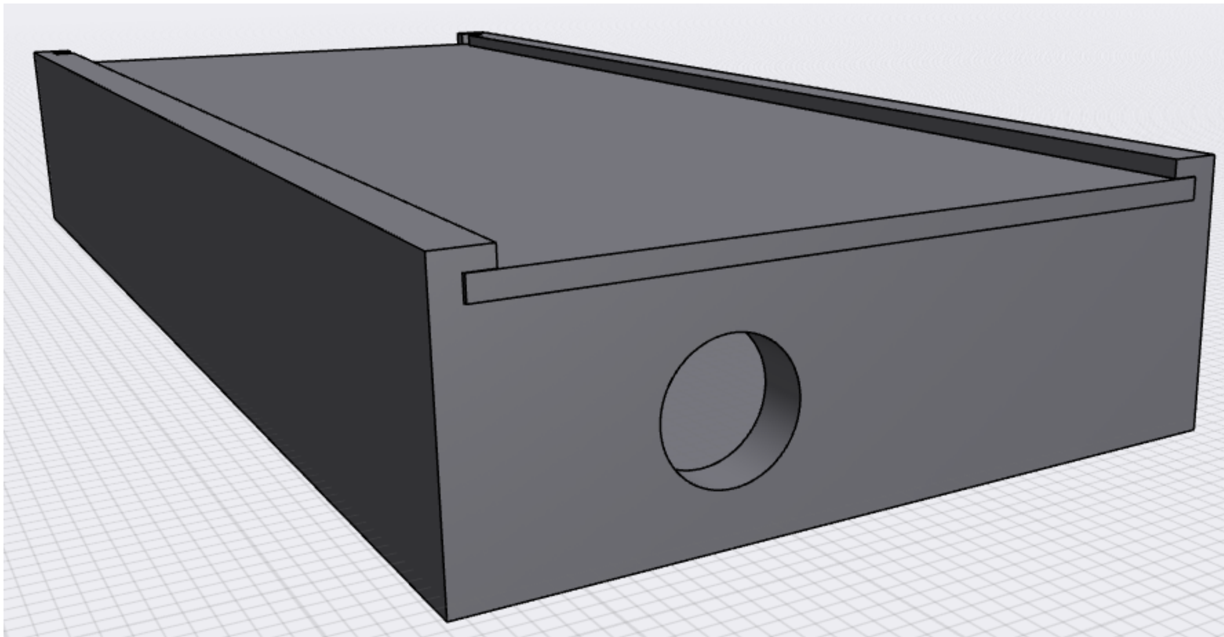

Figure 17: Hole in the enclosure for antenna connector.

After the design, the enclosure was then 3D printed using an Ultimaker 2. In terms of material, the enclosure is made of transparent PLA, which allows the LED to be conveniently seen through the enclosure and eliminates the need for any additional holes in the casing.

Figure 18: The device is very portable and easily held in hand.

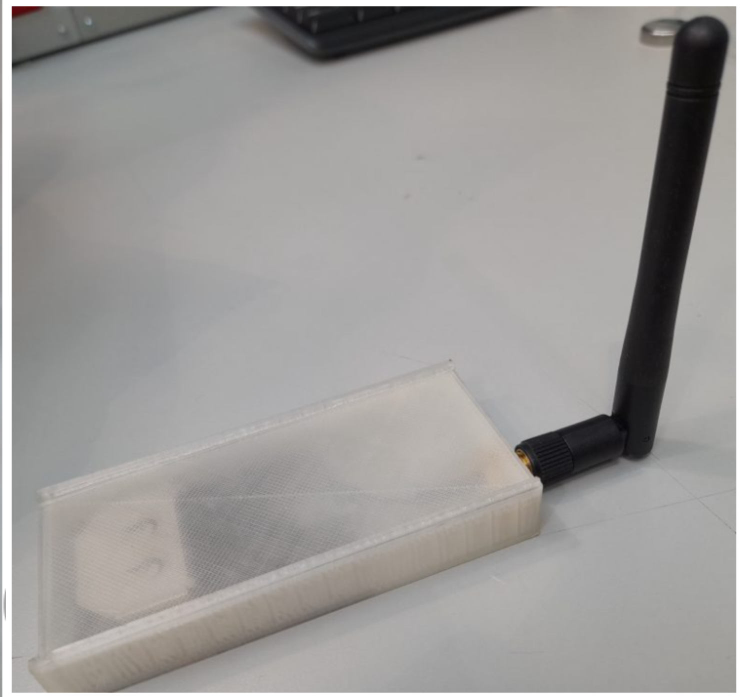

Figure 19: The full device with the enclosure, antenna and the PCB inside.

6. User Experience

In terms of practical user experience, the positives and possible negatives with the device are highlighted in the table below.

Features providing positive user experience	Features possibly taking away from the user experience
Ease of use: Everything inside the device can be slid out very easily if needed. All the user has to do is insert a battery and the device will start functioning.	No off-switch: Currently, the only way to turn off the device is to slide out its battery. This, however, is made very easy due to the enclosure design.
Portability: The device is handheld and very light, which makes it extremely convenient to carry around.	The device's aesthetics: So far, we did not put much thought into making the device look aestheticly pleasing.
Variety and flexibility in alerting methods: The device uses multiple alerting methods, which allows it to be used in various different environments.	Digital interface for Bluetooth functionality: Currently, there isn't any dedicated app for receiving the Bluetooth alert notifications.
Customizibility: The device's firmware can conveniently be customized for specific various use-cases through its wireless DFU functionality.	False drone detections: Since the device's algorithm currently is not able to distinguish 2.4 GHz signals coming from drones and other devices, this can cause issues in an environment with many other electronic devices.

7. Reflection of The Project

7.1. Reaching Objective

All in all, the project our team views the project a success as a whole. The main objectives we were able to reach can be presented as follows:

- Detecting 2.4 GHz Drones (RF Method): After the testing process 2.4 GHz device signal including the drone can be detected by the device.

- Minimum 20 m Detection Range: After testing in the area with the lowest signal noise that we could find, we ensure that the device can catch signal from at least 100m range.

- High Portability: The final product is small and light and can be carried around easily.
- Versatile Alerting Options: There are three ways of alerting which are via sound, light, and Bluetooth Low Energy beacon.
- Low Production Costs: Total cost of all components less than 40 € (not including PCB manufacturing and included VAT).
- Suitable for Military Use: Total cost of all components less than 40 € (not including PCB manufacturing).

7.2. Timetable

The timetable that was made in our original project plan could not be followed in the later phase of the project (from 20th of July) as one of our teammates who was responsible for 3D modelling left the course. Due to this as well as other reasons, we figured that we were not moving fast enough and that we might be unable to finish the project in time. Therefore, we have had to adjust the plan and increase our workload in order to produce the best result that we can. Because of this, we were somewhat unclear in our plan previously and could not make another relevant timetable. However, we were still able to finish the project in time and complete everything that was required.

7.3. Risk Analysis

In the final few weeks of the course, we realized that time management was the most concerning risk due to our limited knowledge and experience. Because of this, we had to spend a lot more time learning and researching as initially planned. To mitigate this risk, we had to increase our workload weekly so that we could meet the deadline.

Moreover, there is one risk that we did not initially taken into account, which was the possibility of a team member leaving the project. Because this risk came true, we had to take on all the additional workload of our teammate who left.

Below is a table showing the risks we outlined in our initial project plan. The column "Initial severity of risk" highlights the severity or relevance of the given risk as we initially thought. The column "Relevance of risk in the end" highlights whether this risk actually came true by the end of the course and how relevant that risk turned out to be.

Risk	Initial severity of risk	Relevance of risk in the end
End device not functioning as intended	High	Not relevant - we managed to get the device functioning in the end
Budget going too high	Low	Not relevant - we managed to stay well below our budget limit of 1500e
Running out of time	High	Somewhat relevant - we managed to complete all the essential tasks in time, however, we were not able to implement everything as initially planned
Not enough knowledge / experience in our team	Low	Not relevant - even though there were areas we lacked experience in, we still managed to reach our initial objectives

7.4. Challenges & Troubleshooting

7.4.1. Firmware Challenges

Making the firmware turned out to be a bit tougher than expected. The biggest parts that took up a lot of time were learning how to use the development kit and nRF SDK, along with some other nRF tools. Getting that first bit of code to actually work on the kit was pretty tough, especially for someone like me (Quoc) who's not really experienced in this stuff.

Another big challenge was putting all the different things we wanted the device to do into one piece of software, as the device cannot conduct scanning and beaconing at the same time, so events handling must be implemented.

7.4.2. Hardware Challenges

For me (Ville) who was responsible for the hardware and 3D modelling aspects of the project (circuit and PCB design, enclosure design and 3D printing), the part with most trouble was the PCB assembly. As the electronic components were very small, it was difficult to place them accurately. Also, because all components were surface-mount, it was near impossible to hand-solder some of them, so a reflow oven had to be used. The most troubling component was the Bluetooth module with 47 total pins, which had to be accurately placed in order for it to function correctly. Due to this, the assembly process required multiple tries.

Another aspect that caused some trouble in the hardware part were some components (a diode and LED) that were placed incorrectly due to misread markings in them. However, this was easy to fix by just desoldering the components and placing them again, once we figured out what the issue was using a multimeter.

In the end device, we also found out that one of the MOSFET's was not suitable for use with the buzzer and this greatly reduced the sound coming from it. Unfortunately, we ran out of time to replace that MOSFET by an appropriate one, so the buzzer was not as loud as we wanted it to be in the final device.

7.4.3. Testing Phase Challenges

As the product use case is a battlefield area, it is impossible for us to get the device testing at exactly such an area. We decided that the woods next to the Aalto University campus was the best place to simulate such an area and did our testing there. As a result, the efficacy of the device's drone detection capabilities in its intended environment remains uncertain.

Moreover, the inherent drawback of the RF-based detection method lies in its inability to differentiate between drones and other devices operating within the same frequency range. As the the testing process was constrained to the Aalto campus area, there were still numerous devices emitting signals within the 2.4 GHz frequency range and this posed challenges in isolating drone signals from the ambient radio frequency noise. Due to this, the results obtained from our limited testing environment may not accurately reflect the device's performance in the intended battlefield context.

8. Discussion and Conclusions

In the journey of developing the Saab Overhead UAV Alert Project, we encountered both successes and challenges that significantly contributed to our understanding of drone detection systems, firmware development, and hardware integration. This section serves to discuss the insights gained, lessons learned, and the overall conclusions drawn from our project's progress.

8.1. Key Insights & Lessons Learned

- Firmware development complexity: One of the major takeaways from this project was the complexity of firmware development. Learning to use the development kit, nRF SDK, and other tools posed initial challenges. Overcoming these obstacles allowed us to gain a deeper understanding of how software controls hardware and how different components interact within the firmware.

- Modular approach benefits: Adopting a modular approach in firmware development proved advantageous. Building and testing individual functionalities before integration helped identify and resolve issues more efficiently. It also facilitated unit testing and ensured that each component worked as intended.

- Hardware design considerations: The hardware design phase emphasized the importance of careful component selection and integration. Balancing cost-effectiveness, availability, and functionality led to a well-rounded product. Additionally, understanding the limitations and capabilities of different hardware elements influenced decisions and optimizations.

- Practical applications of electronics and hardware: As we quickly realized, electronic components and hardware devices can work quite differently in the real-world as they should in theory. Due to this, we had to troubleshoot multiple times and learned a lot about the actual real-world uses and limitations of electronics.

- Time management and team dynamics: We recognized the crucial role of effective time management and the potential impact of changes in team dynamics. The departure of a team member required us to adjust our approach and distribute responsibilities accordingly. This experience highlighted the importance of clear communication and adaptability within the team.

8.2. Project Conclusions

Objective Attainment: The project's objectives were largely met, yielding a functional UAV detection device that offers accurate 2.4GHz drone detection, a minimum range of 20 meters (way further actually), high portability, versatile alerting options, and compatibility for potential military applications.

Commercial Viability: Our emphasis on using commercially available components and cost-effective solutions for both hardware and software sets the foundation for potential mass production and commercialization. This approach not only addresses technical objectives but also acknowledges the economic aspects of product development.

Future Enhancements: While the project meets its primary objectives, we envision several areas for improvement in future iterations:

- Detection algorithm refinement: Enhancing the device's detection algorithm could improve accuracy and minimize false positives, enhancing its reliability in complex environments.

- BLE-based software development: Developing a Bluetooth Low Energy (BLE) companion application could enhance user experience by enabling remote control and configuration of the device via a mobile device.

- Enhanced testing implementation and location: Conducting testing in a controlled environment that better simulates battlefield conditions could provide more accurate performance insights.

- Improved product appearance: Focusing on the aesthetic aspects of the device, such as design and materials, could enhance its overall appeal and user acceptance.

- Enhanced usability: Streamlining user interactions, instructions, and setup procedures could make the device more user-friendly, expanding its potential user base.

As a whole, this serves as a valuable learning experience for the team, providing insights into the complexities of developing a cutting-edge detection device as well as into the general aspects of a full engineering project. The project's outcomes contribute to the field of UAV detection and security, potentially offering a practical solution to safeguarding critical infrastructure and urban environments.

The project's success in creating a functional prototype demonstrates the feasibility of designing a cost-effective, portable, and efficient UAV detection system. Our journey in overcoming challenges and achieving objectives reinforces the significance of collaboration, persistence, and adaptation in project development.

The Saab Overhead UAV Alert Project exemplifies our commitment to innovation, problem-solving, and collaboration. As we conclude this phase of the project, we look forward to applying the knowledge and experience gained to future endeavors, where we can continue to contribute to technological advancements and real-world solutions.

8.3. Acknowledgments

We extend our gratitude to Aalto University and Saab for their support and collaboration throughout the project. The guidance and mentorship provided by our professors and teaching assistants were invaluable in steering the project towards its successful outcomes.

9. List of Appendices

Project files:

- Source code
- KiCad design files
- Gerber files
- Enclosure 3D model
- Final presentation slides and poster

10. References

[1] https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7206421/
[2] https://www.asisonline.org/security-management-magazine/articles/2023/05/uncrewed-aerial-systems/security-implications-drones/
[3] https://www.alliedmarketresearch.com/anti-drone-market-A08180
[4] https://www.iwm.org.uk/history/a-brief-history-of-drones
[5] https://infocenter.nordicsemi.com/index.jsp
[6] https://infocenter.nordicsemi.com/index.jsp?topic=%2Fcom.nordic.infocenter.sdk5.v15.3.0%2Fble_sdk_app_beacon.html
[7] https://github.com/NordicSemiconductor/pc-nrfconnect-rssi