Eemeli Pursiainen

DEVELOPMENT OF REMOTE MOTION DETECTION SYSTEM

Bachelor’s Thesis
Faculty of Information Technology and Communication Sciences
Examiner: Katja Laine
January 2024

ABSTRACT

Eemeli Pursiainen: Development of remote motion detection system
Bachelor’s Thesis
Tampere University
Bachelor's Programme in Computing and Electrical Engineering
January 2024

This thesis outlines the development steps of building a wireless embedded system. The objective of this thesis was to find out if it is possible to develop and manufacture a motion detection system with separate units for detecting motion and alerting the user. The system is intended to work as an indirect doorbell, where a motion sensor is used to detect the presence of a human. Commercial solutions already exist but they lack customizability, which is what the system in this thesis is designed to address.

This thesis describes the thought process taken at each development step with a focus on the specific minor details. It is not a guide, but it rather describes and justifies the choices made. A key takeaway of this thesis is that embedded systems development is extremely detail oriented and it is often the smallest details that account for the success or failure of a development project.

The development process begins with the specification of the system, where realistic expectations for the system are presented. Next, electronic components and modules are chosen to attempt to meet these design specifications. After the important phase of component selection, the components are grouped together and placed in a schematic. The connections between parts in the schematic are used to design a printed circuit board layout, based on which the board can be manufactured. After the circuit board is manufactured, a case can be designed and manufactured to protect the board from damage during programming and testing. The manufacturing of the circuit board also allows for the microcontroller on the board to be programmed and its functionality validated. The system is now functional and can be tested and the results can be compared against the specifications set at the beginning of the process. In product development, the design would be iterated upon, but the scope of this thesis limits the development process to a single iteration.

The end result of the development presented in this thesis yielded a functional system, which partially met the design specifications. The system achieved its design goals in terms of battery life and robustness of the wireless system, whereas the wireless range and motion sensor require further development. The experimental use of a new type of motion sensor ended up being impractical for this application. The sensor provided false positives based on environmental factors, such as rain and heavy wind. Potential solutions to these shortcomings could be the use of external antennas and the use of a different type of motion sensor.

Keywords: LoRa, STM32, Arduino, PCB design, embedded system design, 3D printing.

The originality of this thesis has been checked using the Turnitin Originality Check service.

TIIVISTELMÄ

Eemeli Pursiainen: Langattoman liiketunnistinjärjestelmän kehitys
Kandidaatintyö
Tampereen yliopisto
Tieto- ja Sähkötekniikan kandidaatinohjelma
Tammikuu 2024

Työn aiheena on liiketunnistimella toimivan langattoman ovikellon kehitys vaiheittain tuoteideasta toimivaksi järjestelmäksi. Järjestelmään kuuluu kaksi erillistä laitetta, lähetin ja vastaanotin. Lähetin sisältää liiketunnistimen ja se asetetaan paikkaan, jossa liikettä halutaan valvoa. Vastaanotin sisältää kaiuttimen, joka ilmaisee, kun liikettä on tunnistettu. Laitteiden välinen tiedonsiirto tapahtuu langattomasti. Työssä tutkitaan, onko mahdollista suunnitella ja rakentaa liiketunnistimella toimiva langaton ovikello.

Työssä selitetään ja perustellaan jokaisessa vaiheessa tehdyt valinnat. Selitykset keskittyvät välttämättömiin pieniin yksityiskohtiin, joita ilman järjestelmä ei olisi toiminut. Työ ei ole suoraan kopioitava ohje, vaan se toimii yleisenä ohjeistuksena vastaavanlaisten mikrokontrolleriprojektien valintojen tekoon. Yksi työn tärkeimmistä havainnoista on, että sulautettujen järjestelmien kehityksessä on oltava erittäin tarkka yksityiskohtien suhteen, sillä usein projektin onnistuminen on riippuvainen pienimmistä teknisistä yksityiskohdista.

Langattoman ovikellon kehitysprosessi alkoi järjestelmän ominaisuuksien tavoitteiden määrittämisellä. Tavoitteiden perusteella valittiin piirin komponentit. Kun kaikki komponentit oli valittu, voitiin niiden väliset liitokset määritellä piirikaavioon. Piirikaavion avulla luotiin piirilevyn piirros, jonka avulla piirilevyt valmistettiin. Valmistuksen jälkeen piirilevyihin juotettiin komponentit ja ohjelmoitiin mikrokontrollerit. Tämän jälkeen järjestelmän toimivuus vahvistettiin osa kerrallaan. Kun piirilevy oli vahvistettu toimivaksi, suunniteltiin ja valmistettiin sille sopiva suojakotelo. Tämän jälkeen koko järjestelmä testattiin ja tuloksia verrattiin prosessin alussa asetettuihin tavoitteisiin. Normaalisti tuotteen kehitysprosessissa järjestelmän eri vaiheita parannettaisiin ensimmäisen prototyypin jälkeen, mutta tämän työn laajuus rajattiin prosessin ensimmäiseen iteraatioon.

Työn kehitys tuotti toimivan järjestelmän, joka osittain saavutti suorituskykytavoitteet. Järjestelmän akunkesto ja langattoman yhteyden luotettavuus saavuttivat tavoitteet. Sen sijaan langattoman yhteyden etäisyys ja liiketunnistinanturin tuottamat tulokset eivät saavuttaneet odotuksia. Työssä käytetty uudenlainen liiketunnistinanturi ei tuottanut haluttuja tuloksia, koska se tunnisti ihmisen liikkeen lisäksi myös ympäristöllisiä liikkeitä. Järjestelmän kehittämisen jatkoehdotuksina voisivat olla parempien antennien ja erilaisen liiketunnistinanturin käyttö sekä lähettimen koon pienentäminen.

Avainsanat: Sulautetut järjestelmät, piirilevysuunnittelu, STM32, Arduino, LoRa

PREFACE

The motivation to build the system presented in this thesis came from the lack of a doorbell in the house I moved into recently. Installing a normal doorbell would be the obvious choice, but I prefer to do things my own way.

All schematics and PCB layout images in this thesis have been produced by me specifically for this project using EasyEDA.

Tampere, 7 February 2024

Eemeli Pursiainen

TABLE OF CONTENTS

LIST OF SYMBOLS AND ABBREVIATIONS

Abbreviations

ADC	Analog to Digital Converter
DAC	Digital to Analog Converter
GUI	Graphical User Interface
HAL	Hardware Abstraction Layer
IC	Integrated Circuit
IIC	Inter-Integrated Circuit, digital communication bus
IO	Input and Output
Li-Ion	Lithium Ion, Battery technology
LL	Low-Level
LoRa	Long Range, wireless technology
OLED	Organic Light Emitting Diode, type of display
PCB	Printed Circuit Board
PIR	Passive InfraRed, type of motion sensor
SPI	Serial Peripheral Interface, digital communication bus
WUT	Wake Up Timer

Symbols

A	Ampere, unit of current
Hz	Hertz, unit of frequency
V	Volt, unit of voltage
W	Watt, unit of power
Ω	Ohm, unit of resistance

1. INTRODUCTION

The doorbell is such a ubiquitous device that it is used everywhere on earth, and everyone knows what it is and how it is used. The concept of the doorbell in its entirety is that people can announce their presence. Sometimes, however, it would be convenient to know of someone's presence before they make themselves noticed. What if there was a way to detect the presence of people. The combined use of a motion sensor with a visual or auditory indicator can be used as a sort of indirect doorbell. An approaching person's presence is indicated without requiring deliberate interaction. Making this concept wireless, where the sensor and indicator do not need to be tethered by wires, would allow for great flexibility. The development of such a system is the subject of this thesis.

Commercial solutions of remote motion detectors are readily available, often however with various shortcomings. Commercial motion detection systems are limited by wireless range, battery life, battery type, range of motion detection and customizability. These shortcomings are the motivation for creating a custom solution, as well as a desire to utilize a new type of motion sensor.

In the last five years, microwave human motion detection sensors have become more available to the consumer hobbyist market. What makes these sensors special is that they operate very differently to the common Passive InfraRed (PIR) motion sensors. Traditional PIR sensors measure changes or differences in infrared light levels to detect movement [1]. These new microwave sensors detect movement by transmitting electromagnetic waves in the microwave frequency range and subsequently analyzing the frequency response. The most apparent difference between the two types of motion sensors is that PIR sensors detect movement of any object with a non-ambient temperature, whereas microwave sensors are advertised as detecting only human movement. Another key difference between the two sensor types is that microwave sensors are advertised as being able to detect movement through objects, meaning that they are not limited by the line of sight of the sensor.

Most commercial systems alert users with a doorbell chime. This approach is used for its simplicity for the end-user. In more advanced use-cases there is room for improvement. In addition to an auditory indication, the system could be visual, such as a light or display. In addition to indication of presence, the system could control relays to turn on lights, fans, or other electrical devices. The display could show diagnostic information, such as the amount of times motion has been detected in a given span of time, battery status, and wireless signal strength. The use of a custom, forward thinking design approach allows for the device to be adapted relatively easily to many different use-cases with different sensors. The goal of this thesis is to find out “is it possible to design, build, and test a modifiable wireless system with a motion detection sensor?”

The chapters of this thesis have been laid out in a progressive manner such that readers of any technical level are able to understand how the end results are achieved. The basic theory chapter (chapter 2) provides background information on basic concepts of embedded and wireless systems. System architecture (chapter 3) describes the design specifications of the system. The building process chapter (chapter 4) outlines the assembly of the Printed Circuit Boards (PCBs) and 3D printed cases and the subsequent challenges that were faced during the process. The programming chapter (chapter 5) describes the programming of the system on a high, conceptual level, and highlights a few of the most important technical low-level programming things. The content of the testing and evaluation chapter (chapter 6) is not only limited to the testing of the system, but it also includes an analysis of the achievement of the design specifications set in the system architecture chapter. The conclusion chapter (chapter 7) presents a retrospective look at the project overall, outlining potential future improvements for the system.

2. BASIC THEORY

Logic circuits are used to receive and transmit data signals. Some logic circuits use a sequence-based approach, the core logic element of this approach is a processor. The processor sequentially steps through a predetermined program, performing mathematical and logical operations. To store instructions and other data, the processor is accompanied by at least one memory element. The processor requires a peripheral interface to handle input and output signals to and from external logic components, referred to as a general-purpose Input and Output interface (IO). This combination creates a system which is able to programmatically interface with digital signals. A microcontroller is an Integrated Circuit (IC), which combines a processor, memory, and IO within a single package [2]. An integrated circuit, more commonly known as a chip, is an electronic circuit encased in a package. This compact package allows easier integration into a space efficient design.

Additional peripherals can be added alongside the core components. Manufacturers add additional peripherals to microcontrollers to suit a broad range of use-cases [3]. Some of the most common peripherals are Analog to Digital Converters (ADC), Digital to Analog Converters (DAC), and peripherals that natively handle digital communication protocols. An illustration showing the separate elements which make up a typical microcontroller can be found in Figure 1. ADCs are used to digitally sample a voltage in a designated range [4], most commonly between zero and the operating voltage of the microcontroller. A DAC is the reverse of the ADC. A DAC is used to convert a digital value to an analog voltage [5], where the voltage is limited to the operating voltage of the microcontroller. Digital communications protocol peripherals simplify interfacing with other ICs. Common digital communication protocols include USB, SPI, RS232, IIC and RS485 [6]. Using these peripherals, the programmer avoids tedious programming of every communication protocol by simply configuring the peripheral to required needs. Often, these peripherals achieve greater data transmission rates than what could be achieved by the protocol implemented within a software implementation, this is due to the peripheral running in parallel with the processor.

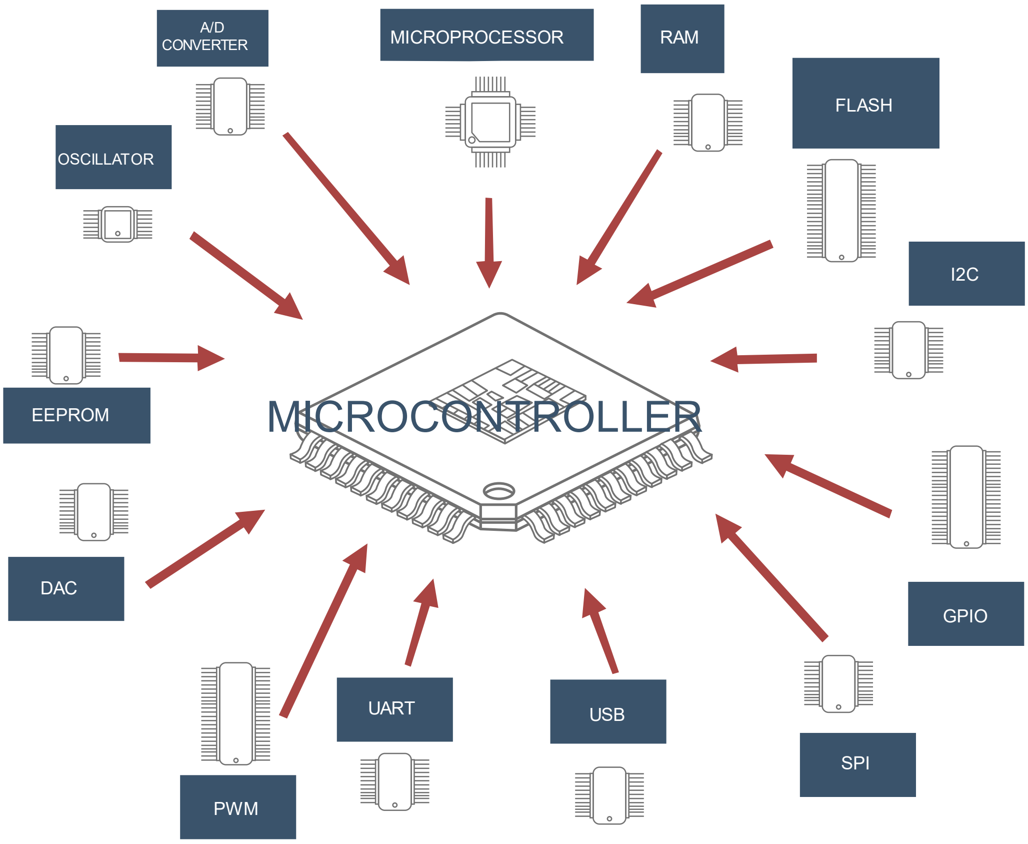


Figure 1. *Elements of a typical microcontroller [7].*

Wireless data transmission is a method to transfer data from one device to another, where electromagnetic waves are used to transmit data without the use of conductors [8]. Electronic circuits are used to synthesize data into signals with frequencies commonly within the radio frequency range. An antenna can emit these signals into the surrounding environment as electromagnetic radiation. For this radiation to be useful, it must be able to be received and decoded back into data. To do this, the reverse of the transmission process is used, where incoming radiation is absorbed by an antenna and turned into an electronic signal. This signal can be sampled by an electronic circuit, which then converts it back into usable data. The advantage of this process is that the two devices do not need to be tethered using conductors.

3. SYSTEM ARCHITECTURE

The system consists of a transmitter unit and receiver unit. The units communicate wirelessly to reduce compromises, allowing ease of system implementation for the end user. Each unit is independently powered with rechargeable Lithium Ion (Li-Ion) battery cells. Wireless communication within the system is performed using a low power, Long Range (LoRa) module on each unit. A microwave motion detection sensor is used to detect movement. The transmitter unit transmits data when movement is detected. Upon receiving data from the transmitter unit, the receiver unit alerts its surrounding area with auditory and visual cues. To exceed the performance of equivalent consumer products, the system was designed with the following minimum specifications.

- Wireless range without obstructions: 200m
- Transmitter battery life with 100 transmissions per day: 3 months
- Receiver battery life: 24 hours
- Removable battery cells
- Motion sensor must avoid false positives from non-human sources
- Reliable data transmission with no lost packets

3.1 Transmitter

The operating principle of the transmitter is very simple. The microcontroller is in a low power mode until the microwave sensor detects movement. When movement is detected, the microcontroller measures the unit’s battery voltage and transmits it to the receiver. The schematic in Appendix 1 shows all necessary components and connections to achieve the wanted operation of the transmitter module.

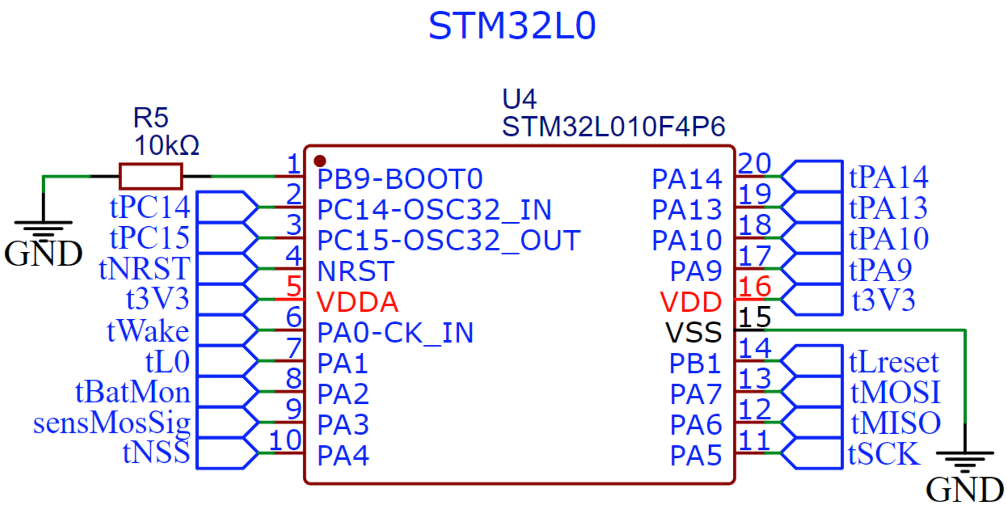


Figure 2. *Transmitter microcontroller pin connections.*

A microcontroller is required in the transmitter unit to fulfill digital communication tasks, checking the status of the motion detector, and measuring the battery voltage. Implementing these tasks in the development phase without the use of a microcontroller would be infeasible.

The microcontroller used in the transmitter is an STM32L010, which was chosen for its low run and standby currents. The microcontroller was also chosen for its hardware implementation simplicity, as can be seen in Figure 2, as it uses very few external components and does not require an external crystal oscillator. The operating voltage of the microcontroller is 3.3V matching the microwave motion sensor and LoRa module. This compatibility simplifies the overall architecture such that only one voltage regulator is required, which reduces sources of quiescent current draw.

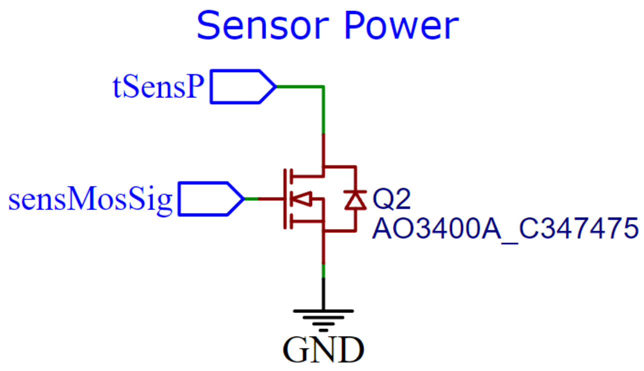


Figure 3. *Transmitter sensor power switching MOSFET.*

Power to the microwave motion sensor is controlled by the microcontroller through a logic-level MOSFET, illustrated in Figure 3. This allows for the sensor power to be disconnected, which reduces current draw when the sensor is not needed.

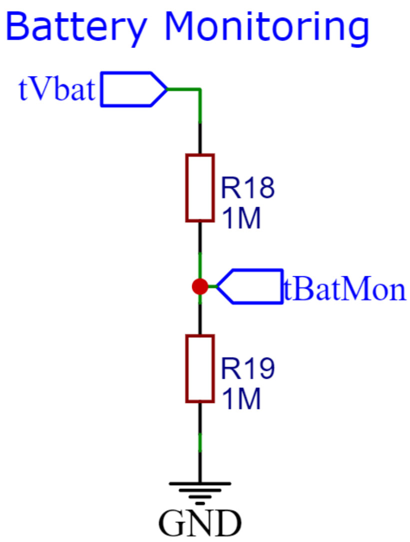


Figure 4. *Transmitter voltage divider.*

A voltage divider with a $\frac{1}{2}$ ratio, as seen in Figure 4, steps down the battery voltage to an appropriate level, such that the microcontroller's ADC is able to sample it. The divider resistors are 1MΩ each, drawing an additional quiescent current of 1.85µA nominally. These resistors can be freely changed if this current proves disproportionately large.

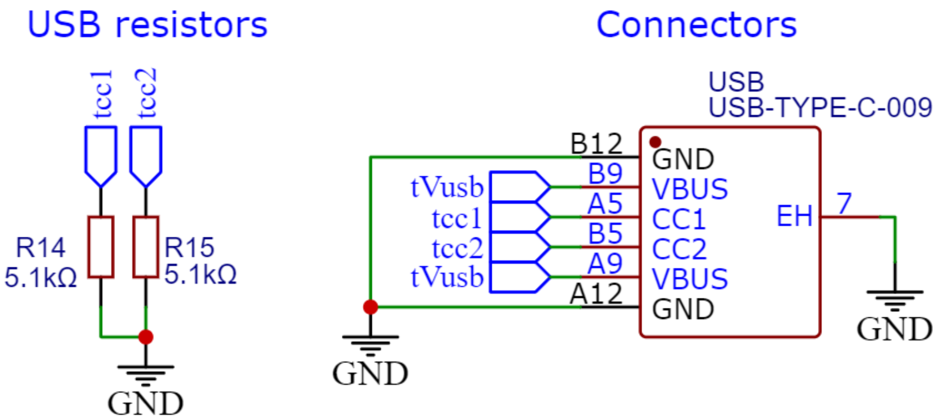


Figure 5. *Transmitter USB connector connections and CC resistors.*

Charging the battery is the sole purpose of the transmitter USB-C connector. The USB-C connector requires two 5.1kΩ resistors on the CC pins to allow for basic USB power delivery functionality. These connections are pictured in Figure 5.

In a battery powered design featuring a low power microcontroller, the quiescent current is the most important aspect of the voltage regulator. The STLQ020PUR from STMicro-electronics features a sub 1µA quiescent current when the output current is below 50µA. A low quiescent current is critical in the design of the transmitter, where the goal is to make the battery last as long as possible. The advertised standby current draw of the STM32L010 microcontroller is 230nA. The relationship between quiescent current and output current of the regulator is illustrated in Figure 6. Based on this graph, the quies-cent current of the regulator is about 250nA when it is supplying 230nA of output current. Currents this low are negligible compared to the other battery management components and the microwave motion sensor.

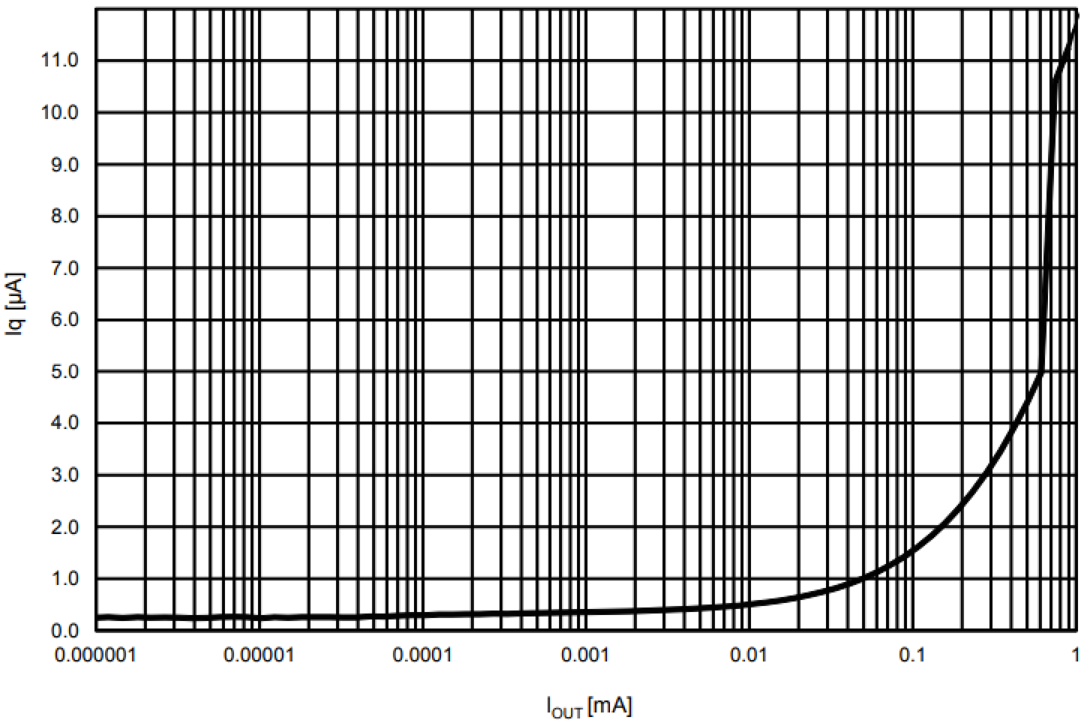


Figure 6. *The voltage regulator Quiescent current of STL020 [9].*

The STLQ020PUR used in this design is an externally configured variant, meaning an external resistor divider is used to set the output voltage to 3.3V. The formula for the resistor divider is $V_{OUT} = V_{ADJ}(1 + R_1/R_2)$, where V_{OUT} is the output voltage and $V_{ADJ}$is an internal reference voltage of 0.8V. The datasheet recommends a total current flow across the two series resistors of the divider to be greater than 500nA. Assuming R_2 to be 1, the resistor divider ratio $\frac{R_1}{R_2} = \frac{3.125}{1}$. Stable output voltage was found with resistor values of $R_1 = 6.65M\Omega$ and $R_2 = 2.15M\Omega$, where the resulting divider current is 375nA. The voltage regulator along with its supporting components are pictured in Figure 7.

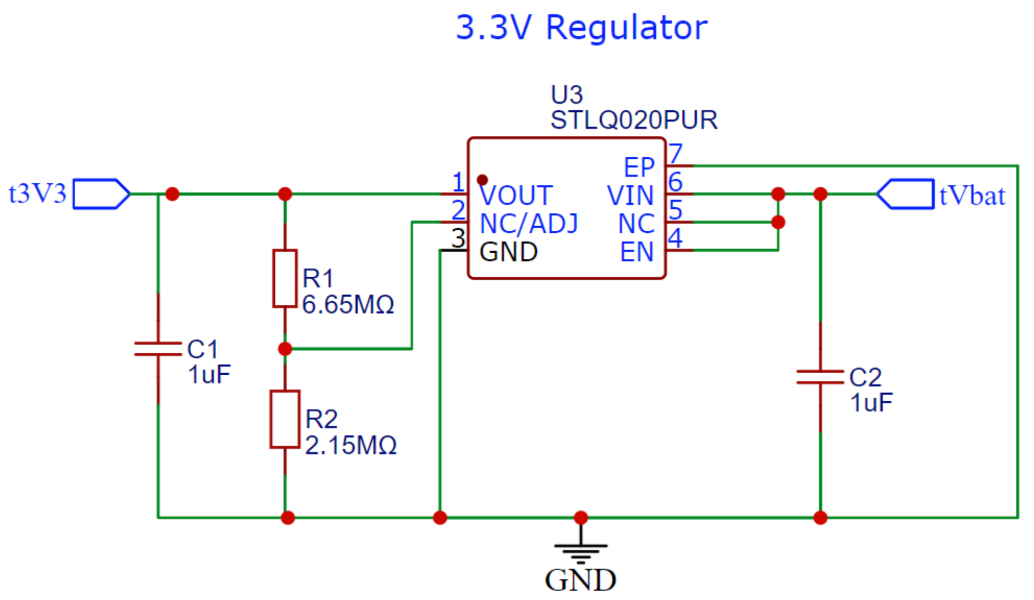


Figure 7. *Transmitter voltage regulator schematic.*

3.2 Receiver

The receiver circuit is more complex than the transmitter for three main reasons. The microcontroller used in the receiver unit has more IO pins and requires more supporting components to function. The receiver unit has to support more features and connect to more interfacing devices than the transmitter. The receiver unit has all remaining IO pins exposed to support future adaptation of additional functions. The full schematic of the receiver unit can be found in Appendix 2.

The current draw of the receiver unit is drastically higher, as the LoRa module is constantly in receive mode. In receive mode, the LoRa module draws about 10mA. The receiver unit is intended to remain connected to a USB power source receiving constant power, thus circumventing the higher current draw.

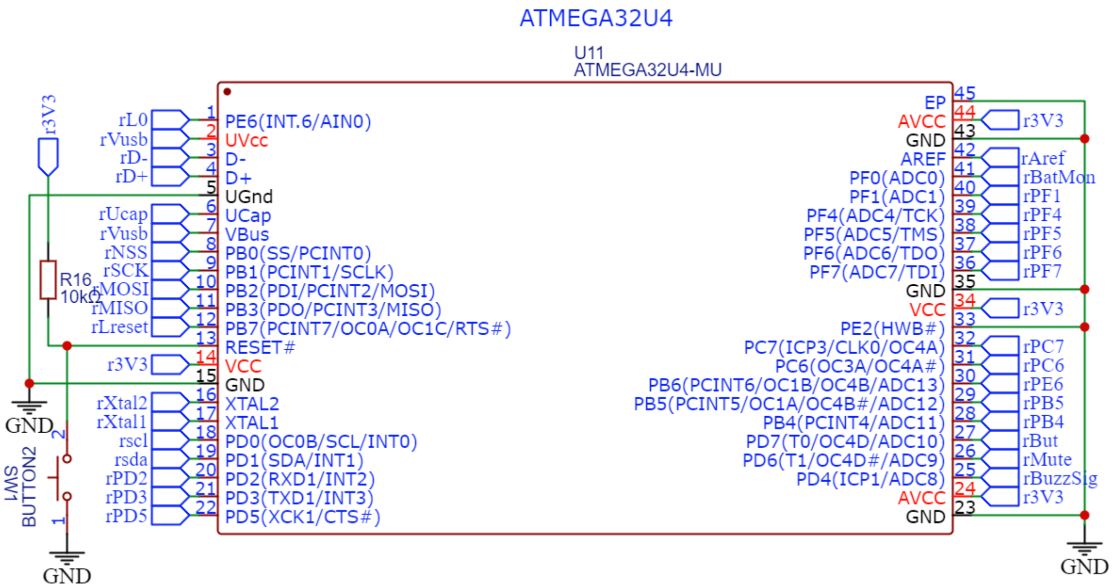


Figure 8. *Receiver microcontroller pin connections.*

Similarly, to the transmitter unit, the receiver unit requires a microcontroller to digitally interface with the LoRa module using SPI and the Organic Light Emitting Diode (OLED) display using IIC. All pin connections for the microcontroller are shown in the schematic in Figure 8. The ATmega32u4 microcontroller was selected for use in the receiver unit for its built-in USB capability and its current market availability. Another significant benefit is its compatibility with the Arduino platform, which simplifies programming.

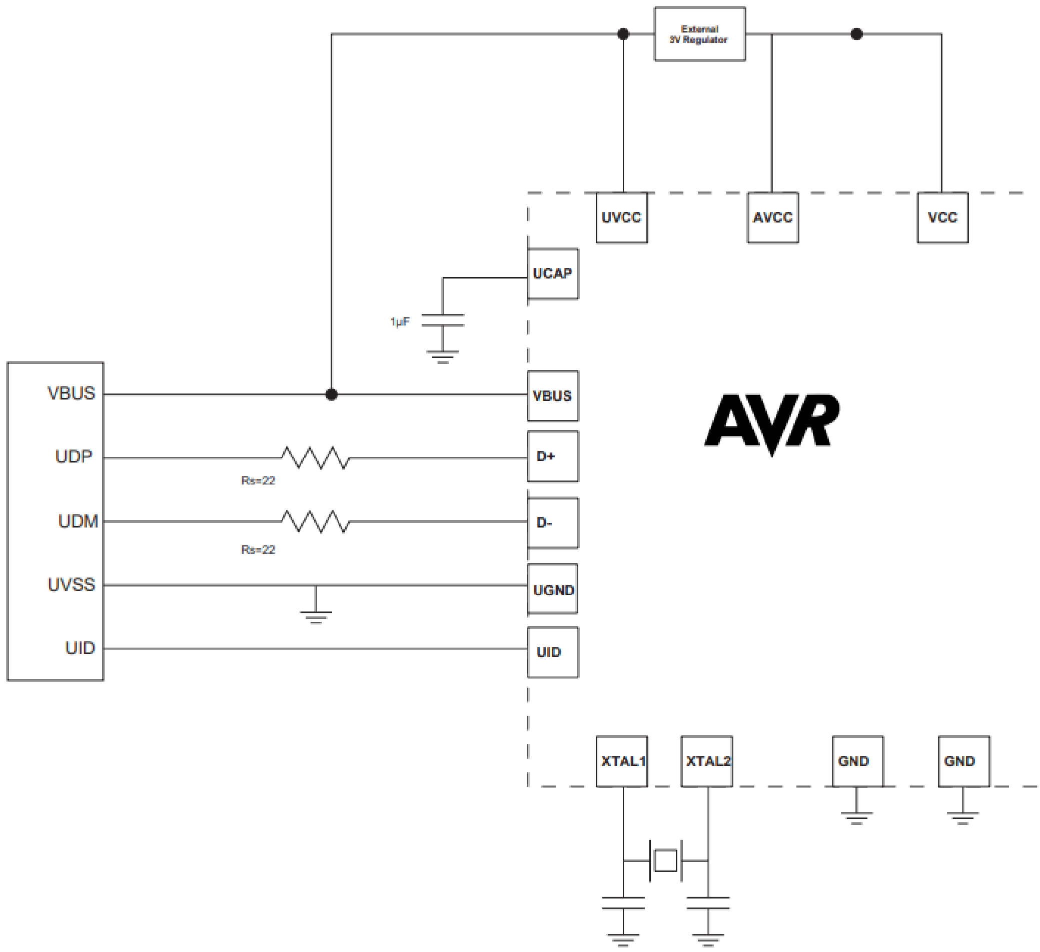


Figure 9. *ATMega32u4 pin connections for using USB when VCC is 3.3V [10].*

While the operating voltage of the microcontroller is 3.3V, it must be duly noted that the VBus and UVcc pins must be connected to 5V coming from the USB [10]. An example of this connection can be found in Figure 9. The receiver unit's USB-C port is versatile, as it is used to charge the battery, communicate with the microcontroller via USB, and program the microcontroller.

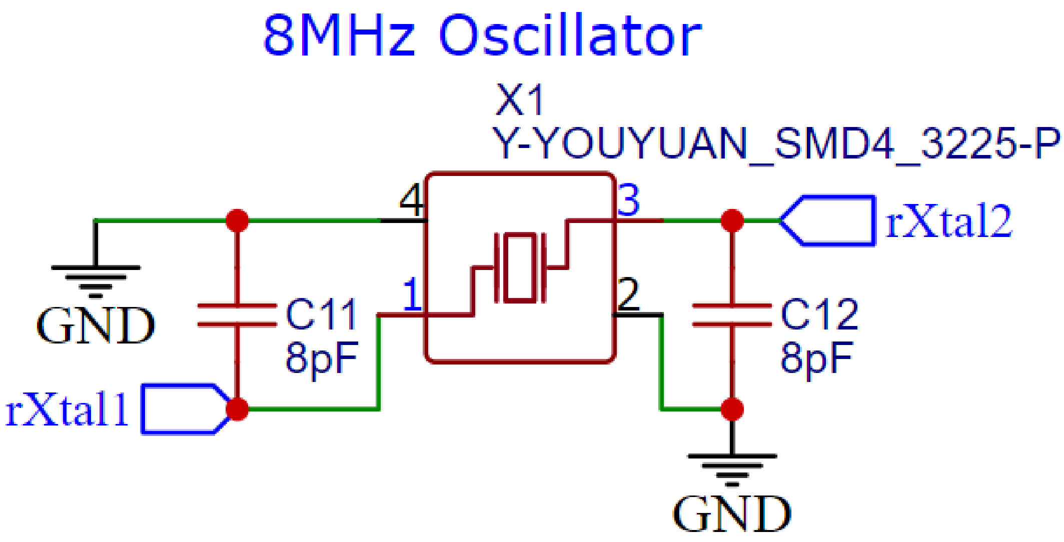


Figure 10. *Receiver microcontroller crystal oscillator with its loading capacitors.*

The ATmega32u4 requires an external 8MHz crystal oscillator to run at 3.3V. The crystal oscillator requires two loading capacitors C11 and C12, which is shown in figure 10. The values of the capacitors are determined by the manufacturer and given in the datasheet of the crystal oscillator.

The receiver unit PCB includes provisions in the form of pin headers for an IIC OLED display, multi-position slide switch, buzzer, and an auxiliary pin header. The auxiliary pin header exposes all remaining GPIO pins of the microcontroller for potential future use. A logic level N-channel MOSFET drives the buzzer, as the microcontroller IO pin source and sink currents are insufficient to drive the high current of the buzzer. The battery voltage is measured with an identical voltage divider as with the transmitter unit.

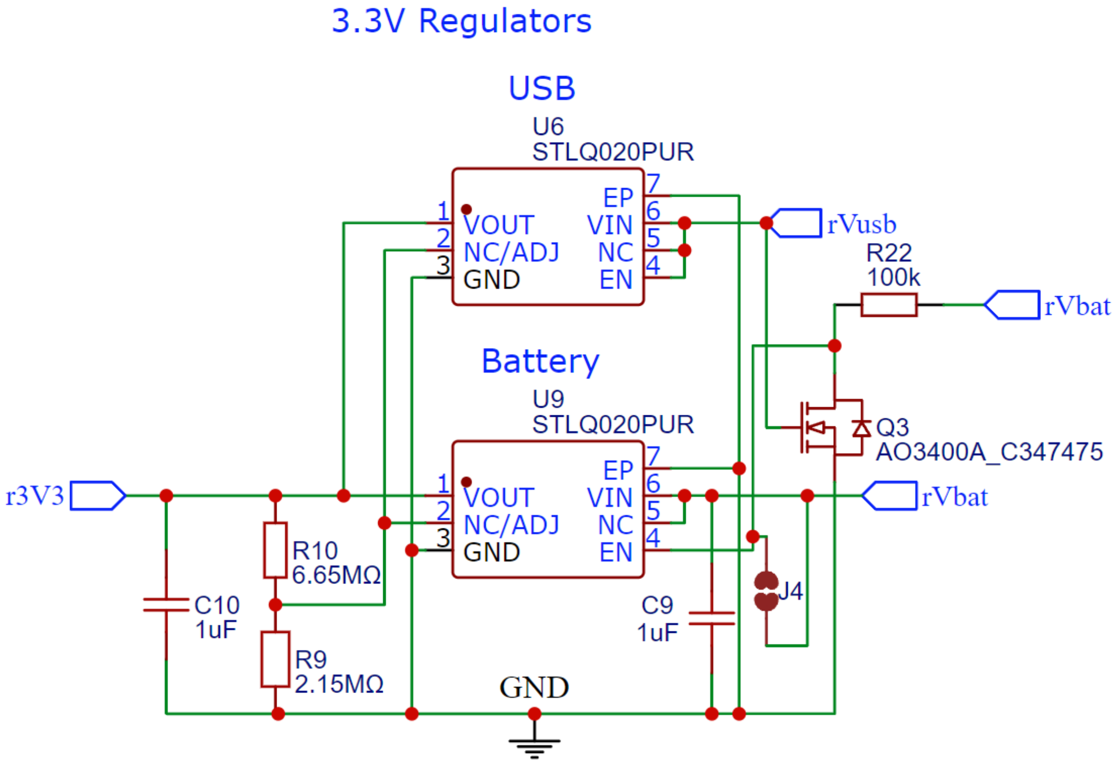


Figure 11. *Receiver voltage regulator schematic.*

The battery portion of the voltage regulator is identical to the transmitter unit. As the receiver unit is intended to be powered from primarily USB, the addition of a second voltage regulator allows the circuit to be powered via USB while it is connected. A logic level N-channel MOSFET is used as a digital inverter. This allows the battery voltage regulator to be disabled while the USB voltage regulator supplies power to the receiver unit circuit. The connections for the regulators and their supporting components can be seen in Figure 11.

3.3 Common parts

Both the transmitter and receiver units are battery powered and have similar peak current draw. This means that they can share the same battery protection, charging and voltage regulation design.

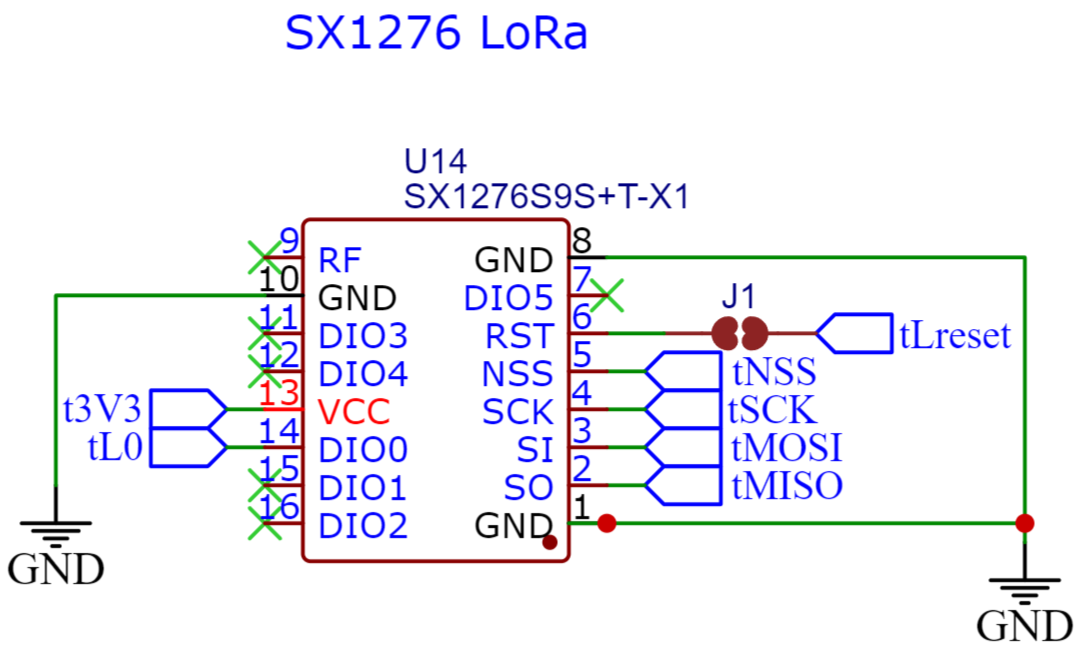


Figure 12. *LoRa module pin connections.*

A wireless module is required in the transmitter unit to wirelessly transmit small amounts of data. A LoRa wireless module is an ideal choice for this application because it allows for low power long distance wireless communication along with easy software implementation. The pin connections for the LoRa module are pictured in Figure 12.

The LoRa module used in this design features a Semtech SX1276 IC. This module was chosen due to its current availability, pin compatibility with other modules, low sleep current and higher transmission power than some other LoRa modules. The SX1276 module used in this design is interchangeable with an RFM95W LoRa module if the availability of SX1276 modules were to change. The SX1276 can be set to operate at a frequency of 868MHz, which is a permitted transmission frequency within the EU [11]. The sleep mode current of the SX1276 is 200nA. This functionality circumvents the need to switch off power to the LoRa module externally.

An important detail to note about connecting the LoRa module is the solder jumper used to disconnect the RESET pin from the microcontroller. The connection of the RESET pin is redundant for the SX1276, however some other LoRa modules may require its use. If left connected when not required, the RESET pin can cause an undesired excess current draw in sleep mode. This is due to some LoRa modules driving this pin low or high in sleep mode causing excess current to be drawn.

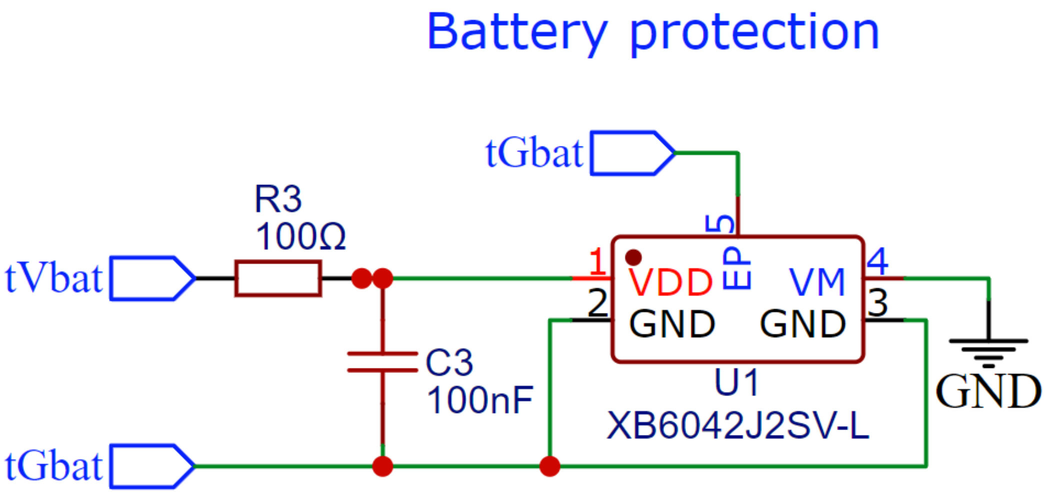


Figure 13. *Battery protection schematic.*

Both units' battery cells are protected with XySemi XB6042J2SV-L Li-Ion protection circuits. Important protection features for Lithium batteries are overcharge, over-discharge, overcurrent, and temperature protection. Ease of implementation, low 1.5μA quiescent current, and small DFN-4 1x1mm package led to this IC choice. Package size was not important for this application but was chosen to demonstrate hand soldering feasibility for more compact applications in the future. The accompanying passive components used with the IC are taken directly from the datasheet. The battery protection IC and its supporting components are shown in Figure 13.

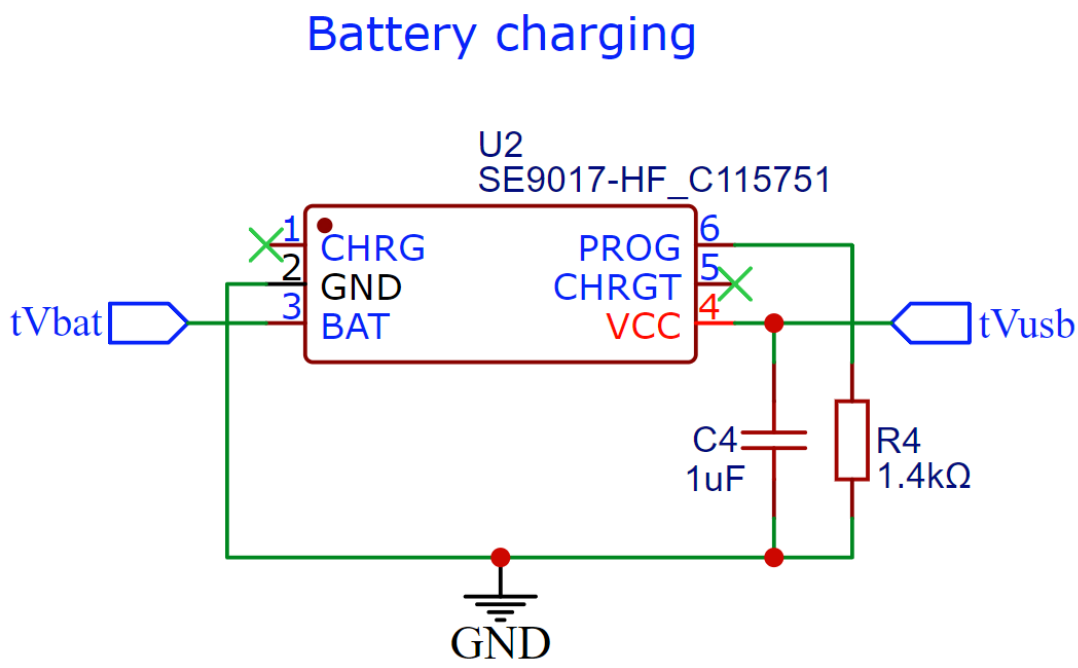


Figure 14. *Battery charging schematic.*

To extend the lifetime of Lithium battery cells, charge current is matched with the state of charge according to a predetermined graph. IC manufacturers have developed easy to use ICs for this purpose. The battery charging IC is the Seaward Electronics SE9017-HF. This IC was chosen for its small SOT-23-6 package and maximum charging current of 800mA. The formula for the maximum charge current resistor $R_{PROG} = \frac{1060V}{I_{CHG}}$. The charge current chosen for this design is 750mA, making $R_{PROG} = 1.4k\Omega$. The schematic for the battery charging IC is shown in Figure 14.

4. BUILDING PROCESS

The building process is split into two logically separated sections. The first describes PCB layout and soldering of components and the second describes the design and assembly of the 3D printed cases.

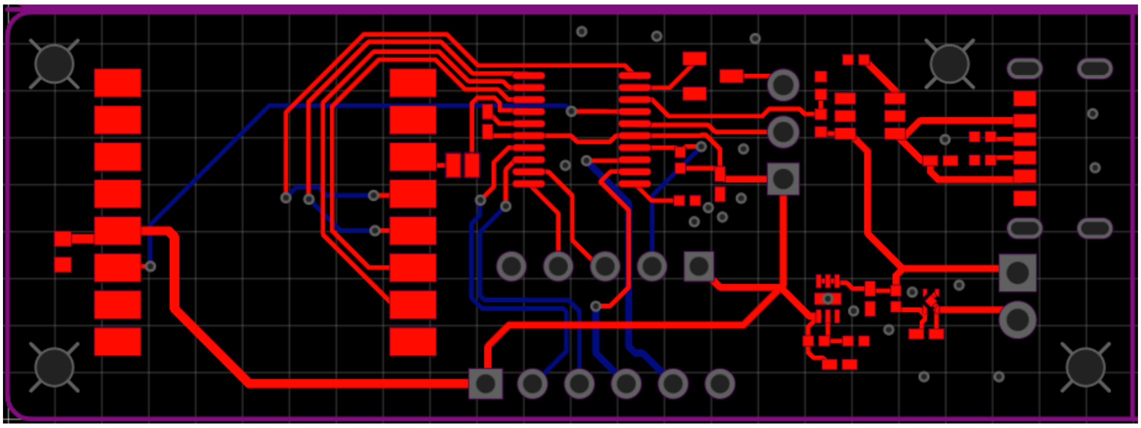


Figure 15. *Layout of transmitter unit PCB.*

The transmitter board components are dispersed, making trace routing straightforward. The PCB layout for the transmitter board is pictured in Figure 15. The PCB footprint was drafted to be roughly the size of the unit’s battery cells i.e., an “18650” cell.

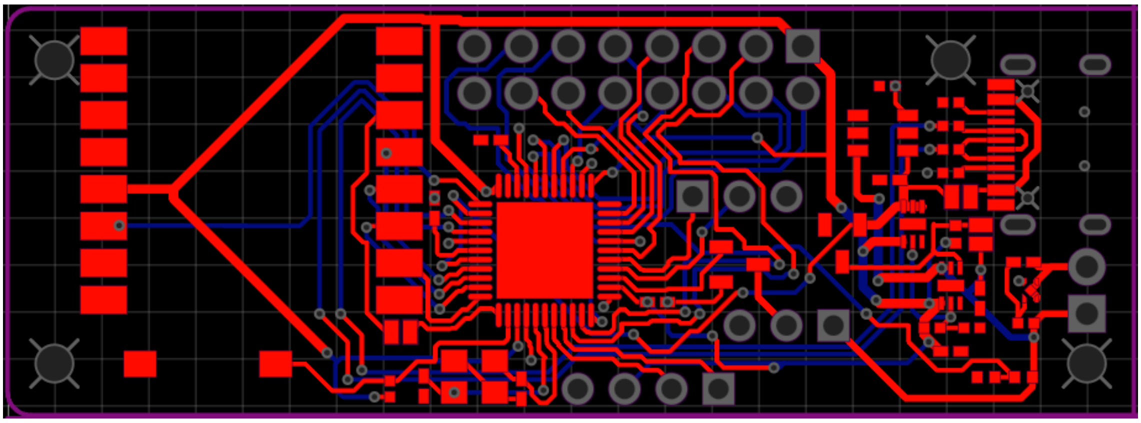


Figure 16. *Layout of receiver unit PCB.*

The design of the receiver board is slightly more convoluted than the transmitter. Many traces require compromises in terms of routing to reach their endpoints. The complexity of the receiver unit PCB is shown in Figure 16. Important traces, such as the 3.3V rail and SPI bus were prioritized in routing. The routing of some traces, such as the signal lines going to the crystal oscillator were compromised, but this does not impact the functionality of the circuit. The microcontroller package makes the layout process more challenging, as the center is a ground pad, preventing traces being routed beneath the microcontroller on the top layer.

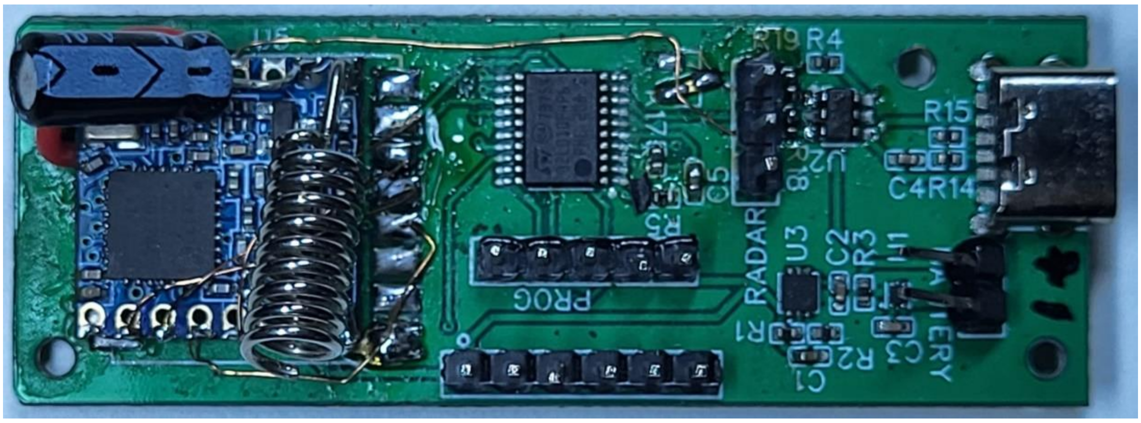


Figure 17. *Working transmitter unit PCB with required modifications.*

The first version of the transmitter unit PCB required a few modifications in the form of enameled copper wire to function correctly. These modifications are pictured in Figure 17. The largest flaw of the first version was the LoRa module footprint not matching the actual LoRa module due to a mismatch of product photos provided by the supplier and the actual delivered product. The only other flaw was the use of an incorrect pin for the WKUP function of the microcontroller. The microcontroller, due to its relatively low pin count, is only able to wake up from standby mode with the use of one specific pin. Unfortunately, the documentation for this functionality is vague to the point where even some STMicroelectronics employees were misled to think that the functionality can be switched to different pins.

The battery management and power supply sections functioned as expected. Even the 1x1mm DFN-4 package was possible to solder by hand with the use of a hot air gun. The USB-C power connection also functioned as expected. The large bypass capacitor is attached directly to the LoRa module as the protruded castellated holes of the module allow for easy soldering.

The first version of the receiver board required the same LoRa footprint modification as the transmitter, as well as rerouting some power traces from the USB 5V rail to the microcontroller. Two 3.3V traces had to be cut and connected to the 5V rail with thin enameled copper wires. After fixing the power traces, the USB communication functioned as expected. These modifications can be seen in Figure 18.

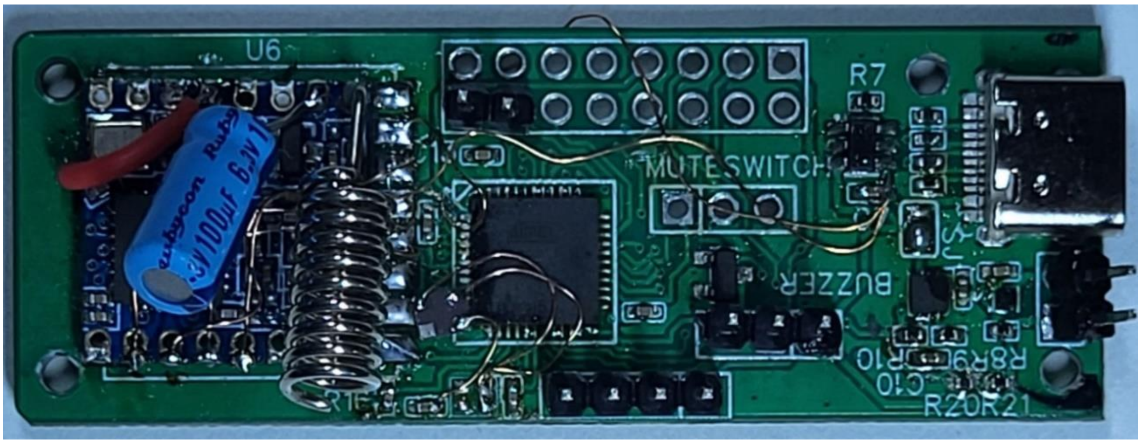


Figure 18. *Working receiver unit PCB with required modifications.*

The cases for the transmitter and receiver were designed in Fusion360. The cases for both units are purely functional. They shield the circuit boards and batteries from damage during system testing. Both units use two 18650 sockets each enabling the use of swappable 18650 battery cells. The cases were 3D-printed using PLA plastic, with a 0.4mm nozzle with a layer height of 0.2mm. The tolerances of the 3D printed parts were specifically designed to suit these dimensions.

Figure 19. *Transmitter module with battery and electronics covers removed.*

The transmitter has a two-piece lid to allow for the changing of batteries leaving the electronics protected. Figure 19 shows the two-piece design of the transmitter unit battery cover. The electronics compartment houses the motion sensor and a power switch in addition to the main PCB. The power switch directly disconnects the battery from the circuit. The screws securing the lid are coarsely threaded self-tapping M2 screws.

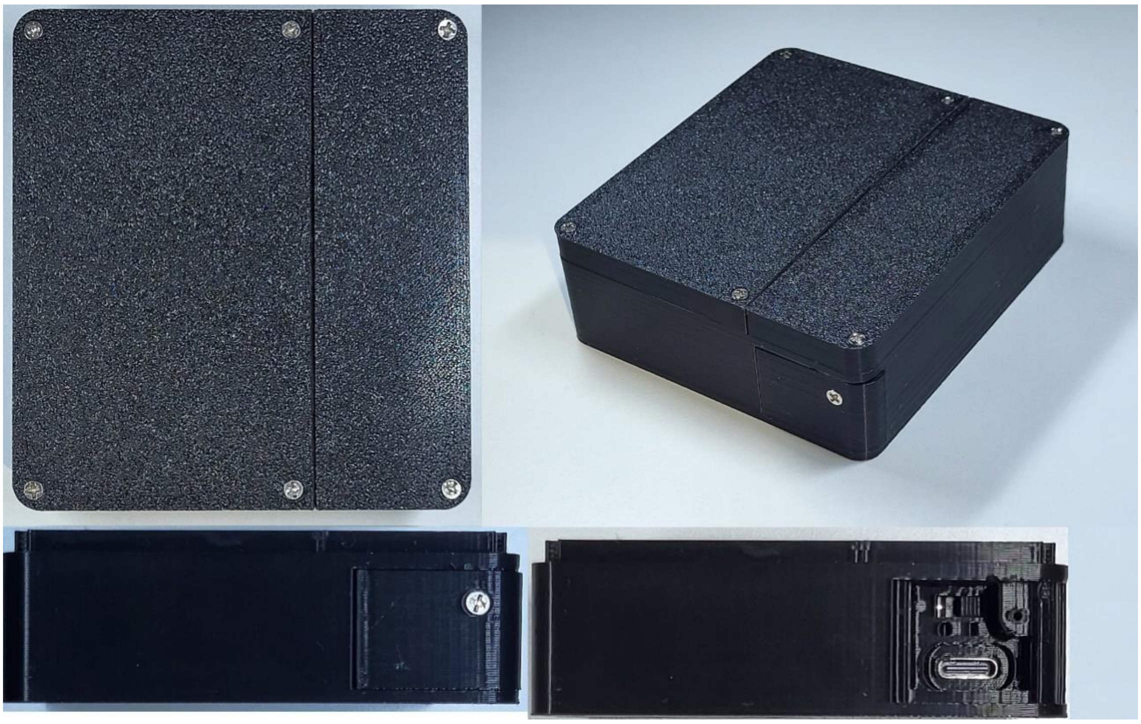


Figure 20. *Transmitter case from different angles.*

The transmitter has a removable cover for the power switch and USB-C port, as it is intended to only be rarely charged. Protection of the unit has been prioritized over charging port accessibility; the port is shielded by a screw-on cover, as illustrated by Figure 20.

Figure 21. *Receiver case from different angles.*

The receiver indicates battery and signal statuses on an integrated OLED display. The receiver has two switches, a main power switch and a mute switch. The mute switch directly disconnects the buzzer from its circuit. This prevents any possible software bugs causing the buzzer to be activated when not intended. The receiver has an RBG-LED to indicate a LoRa packet was received, allowing for silent system operation. All features of the receiver unit can be seen from different angles in figure 21.

5. PROGRAMMING

The transmitter uses an STM32L010 microcontroller with 16KB of flash and 2KB of RAM. The STM32 is programmed using the C programming language using the STM32CubeIDE software development environment. STM32 microcontrollers have numerous configurable registers, allowing for runtime configuration of the internal systems of the microcontroller [12]. Producing a functional program requires meticulous programming of these registers. Fortunately, STMicroelectronics produces a Graphical User Interface (GUI) -based program called STM32Cube to configure basic functionality of the microcontroller. Such basic functionalities are but not limited to clock speeds, IO-functionality, timers, and peripherals.

One drawback of register configuration is the significant portion of program flash lost to store configuration data. This places a major constraint on program size. STMicroelectronics have marginally eased this issue by allowing users to select Low-Level (LL) drivers instead of the typical Hardware Abstraction Layer (HAL) drivers in STM32Cube. The use of LL drivers allows for less flash usage, but as a downside the user must provide implementations for select functions normally included in HAL drivers.

IP	Run/active mode	Sleep mode	Low-power run mode	Low-power sleep mode	Stop mode Wakeup capability	Standby mode Wakeup capability
Consumption V_{DD}=1.8 to 3.6 V (typ)	Down to 140 µA/MHz (from Flash memory)	Down to 37 µA/MHz (from Flash memory)	Down to 8 µA	Down to 4.5 µA	0.29 µA (no RTC) V_{DD}=1.8 V	0.1 µA (no RTC) V_{DD}=1.8 V
					0.54 µA (with RTC) V_{DD}=1.8 V	0.41 µA (with RTC) V_{DD}=1.8 V
					0.34 µA (no RTC) V_{DD}=3.0 V	0.23 µA (no RTC) V_{DD}=3.0 V
					0.67 µA (with RTC) V_{DD}=3.0 V	0.53 µA (with RTC) V_{DD}=3.0 V

Figure 22. *Power consumption of different STM32L010 power modes [13].*

The STM32L is a low power consumption series of microcontrollers, which come with a suite of low power modes. These power modes and their associated power consumptions are shown in Figure 22. When in the lowest power mode, standby, most internal peripherals are powered off including the dynamic memory. Program execution begins from the entry point when recovering from standby, as opposed to the last instruction executed when recovering from other power modes. [13] This means the program execution starts from the beginning each time the microcontroller wakes up from standby mode. The only ways to exit standby mode are via an RTC interrupt or an external pin change interrupt.

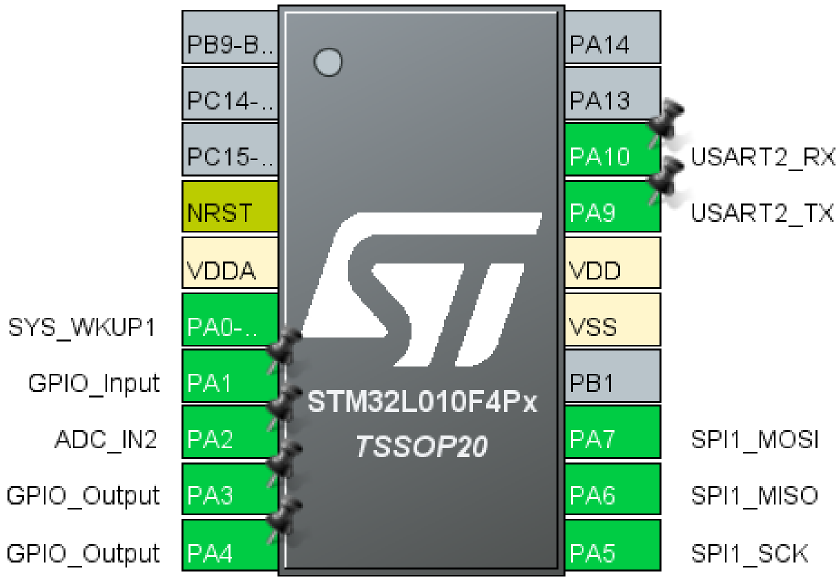


Figure 23. *Selected pin modes on STM32.*

Peripherals on STM32 microcontrollers are software configurable to multiple physical pins. Few select functions such as WKUP functionality can only be accessed via their designated pins. The WKUP functionality is the only method to wake the microcontroller from standby mode via pin change interrupts. As such, it is important to connect the sensor output to the WKUP pin in the circuit schematic, and not assume that the function can be rerouted to a different pin via software configuration. Figure 23 shows pin PA0 configured to act as the SYS_WKUP1 pin.

STM32 microcontrollers have many flags indicating current and previous states of the microcontroller. For example, the microcontroller has flags for each separate type of event, such as waking up from standby via a WKUP pin. These flags can be read by the program, but they are also read by internal functions. Sometimes these unintentionally block desired functionality when set. Flag requirement criteria are vague for certain functions, such as entering standby mode. Therefore, it is very important to clear flags after they have been set.

The STM32 has register modification protection flags, these flags are set by internal and external functions of the program. This can lead to programming errors if not accounted for. These programming errors can be very hard to troubleshoot, as the microcontroller can execute flag clear functions, but they do nothing and do not raise errors if the protection flags are set. The configuration functions generated by STM32Cube occasionally produce such programming errors.

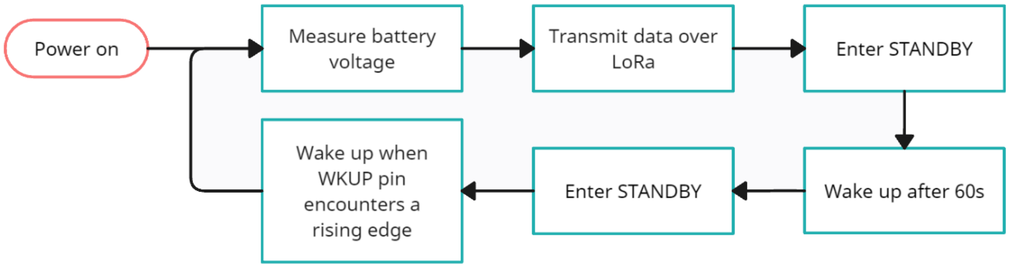


Figure 24. *Transmitter program flowchart.*

The program flow for the transmitter is streamlined, as its only goal is to transmit data whenever it detects movement. The high-level program flow of the transmitter is shown in Figure 24. The microcontroller measures the battery voltage using the ADC. The ADC is configured to a resolution of 12-bits and samples readings over 39.5 clock cycles. The ADC clock is set to 16MHz to match the frequency of the system clock. These settings allow for consistent, accurate and reliable voltage readings.

The LoRa modules are configured to operate at 868MHz with a bandwidth of 31.25kHz. The spreading factor of the modules is set to 7 and the output power is set to the legal maximum of 25mW. [11] With an estimated average of 5 transmissions per hour, this system falls within the standard of 0.1% duty cycle set by EU telecommunications law [11].

```
void sendMessage(char message[10]) {
    loraBeginPacket();
    uint8_t *messageStr = (uint8_t*) message;
    loraWriteBuf(messageStr, strlen(message));
    loraEndPacket(false);
}
```

Program 1. *Structure of the sendMessage function.*

The data transmission over LoRa works by initializing the LoRa module, sending a character array to the LoRa module, and waiting for the data to be sent. The reformatting of the character array message into a pointer seen in Program 1 is not consequential to the operation of the function and could be done prior to this function. The reason for reformatting it inside the sendMessage function is to simplify the formation of the array, as it is treated as a character array everywhere else in the program.

```
   int nackCounter = 0;
2     while (!receiveAck(message) && nackCounter <= 20) {
          if(nackCounter > 0){
4             message[5] = 'L';
          }
6         loraSleep();
          LL_mDelay(500);
8
          sendMessage(message);
10        nackCounter++;
      }
12
      loraSleep();
14    LL_mDelay(100);

16    EnterStandbyModeRTC();
```

Program 2. *While-statement for checking acknowledgement of packet.*

To ensure that the correct message was received, the receiver retransmits the same message back to the transmitter. If the messages match, the microcontroller proceeds to standby mode. If over 500ms have passed since the previous transmission with no reply or the messages do not match, the microcontroller retransmits the message. If a successful transmission has not been made within 20 attempts, the receiver is assumed out of range and the microcontroller goes to standby mode.

After transmitting, the microcontroller is put into standby mode and is configured to wake up after 60s. This is done on line 16 of Program 2. When woken up after 60s, the microcontroller re-enters standby mode, but in this instance, it is configured to wake up from a rising edge of the WKUP pin. Placing the microcontroller into standby for a set amount of time, 60s in this case, allows for greater energy savings during the hysteresis period to avoid multiple detections of the same source of movement. By placing the microcontroller into standby with only a timer to wake it up, any movement detected by the sensor is ignored, as the WKUP interrupt is not enabled.

```
if (LL_RTC_IsActiveFlag_WUT(RTC) == 1) {
        LL_RTC_ClearFlag_WUT(RTC);
        LL_RTC_DisableIT_WUT(RTC);

        loraSleep();

        EnterStandbyModeWKUP();
    }
```

Program 3. *Differentiating between wakeup methods based on state of WUT flag.*

The wake-up timer (WUT) flag is used to differentiate between wake-up sources. The if-statement used to check the WUT flag is shown in Program 3. If the flag is set, the microcontroller was woken up via the WUT. If the flag is not set, the microcontroller was either powered on or it was woken up by the WKUP pin interrupt. If the microcontroller woke up via the WUT, the microcontroller is configured to only wake up via a WKUP interrupt and go back to standby mode. If the microcontroller woke up due to a WKUP interrupt, the programs starts as if the microcontroller was freshly powered on.

The receiver unit uses an Atmel ATmega32u4 microcontroller. This microcontroller is commonly used in consumer development boards, specifically Arduino Leonardo and Micro boards. This means programming is simplified greatly by use of the Arduino IDE and libraries. The ATmega32u4 can be programmed without an external USB to UART converter IC, streamlining the development process.

The Arduino LoRa library was used to simplify communicating with the LoRa module. Modification of the IIC address to 0x3C was required for a working configuration of the OLED with the Adafruit SSD1306 library.

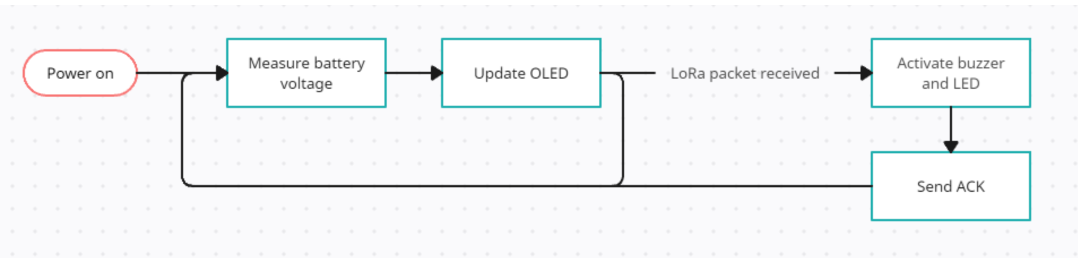


Figure 25. *Receiver program flowchart.*

The program flow, pictured in Figure 25, for the receiver microcontroller is streamlined. While waiting for LoRa packets, the microcontroller measures battery voltage and displays the reading on the OLED display. When a LoRa packet is received, the microcontroller powers the buzzer and LED, then transmits an acknowledgement packet back to the transmitter.

6. TESTING AND EVALUATION

The design specifications for the system were broad and as a result, the testing of the system does not need to be excessively accurate. The tests conducted on the system were intended to validate the general functioning of the system. The most important design specifications were tested separately, and the simplified results of the tests are shown in Table 1.

Table 1. *Key design specifications and their tested results.*

Design parameter	Design specification	Measured result
Transmitter battery life	3 months	18 months
Wireless range	200m	120m
Wireless reliability	Absolute	Absolute within suppressed range
Motion sensor reliability	Absolute	Not reliable

A low idle current draw was a design priority of the transmitter unit. The transmitter unit's idle current draw is 52.3μA. The microwave motion sensor draws most of this current. With the sensor removed, the remaining system draws only 9.5μA. The system draws 5.1mA when the microcontroller is in run mode when the LoRa module is not transmitting. Every time motion is detected, the microcontroller remains in this mode for about 1s. When transmitting, the LoRa module draws about 250mA for a short duration of about 20ms.

An exaggerated scenario for testing transmitter battery life was created by discharging a 520mAh lithium battery cell using transmitter operation beyond reasonable expectation. The transmitter would transmit once every 30s, meaning 2880 transmissions per 24h period. An average of 100 daily transmissions was initially estimated for the system. Exaggerating the transmission rate within the test made testing reasonably quick for development cycles. The 520mAh battery discharged in 45 days. The average energy consumption of the transmitter calculated based on this test is 11.5mAh per 24h period.

The desired battery life of the transmitter was 3 months. The unit's two parallel 18650 Li-Ion battery cells have a total nominal capacity of 6400mAh. Assuming 2880 transmissions per 24h period, the system's theoretical lifespan is approximately 18 months. This lifespan is calculated using the average energy consumption from the exaggerated test. The results of this test greatly exceed the design specifications of 3 months. Given that the transmission rate of the test is nearly 30 times higher than the design specification, the actual transmitter lifespan can be assumed to be much greater than the theoretical estimate.

The system was designed to fit most use-cases by using wireless communications. Thus, the performance of wireless communication was tested. A realistic test scenario was modeled as a mostly unobstructed line-of-sight range test. The antennas used in this test were a coil antenna on the transmitter unit and a quarter wavelength wire antenna on the transmitter unit. Both antennas were housed inside the plastic cases of the units. The system lost no packets transmitting at a range within 120m, beyond which, the system began dropping packets. The system could not communicate reliably beyond 140m.

The wireless communication of the system functioned without errors within the measured 120m range. The design specification of a robust wireless system has been achieved when the range requirement of the system is ignored.

The capabilities of the microwave human motion detection module did not meet its advertised specifications. The testing proved that the module detects the motion of most moving objects. These objects include rain droplets, falling leaves, moving trees and animals. The system was designed to only alert the movement of humans, and as such, the module is not suitable for this application.

7. CONCLUSION

The process outlined in this thesis yielded a working system with room for improvement. Some design specifications were met, and others were not. As with any research and development project, iterations of the design must be made. The system can now be improved based on the tests conducted. In addition to making improvements to this system, the information presented in this thesis can be used to develop future systems more effectively.

Use of different antennas may introduce a significant range increase. A preliminary test comparing the coil antenna provided with the LoRa modules to a quarter wavelength wire antenna showed that the coil antennas placed close to the PCB inside the units significantly decreases the range of the system. An easy improvement for the range of the system can be to add external antennas to both units. Not only are these external antennas likely higher quality in terms of radiation pattern, but also they can be placed away from the PCB, allowing for less obstructions. External antennas can be attached using soldered on coaxial antenna wire to the lora module or added using LoRa modules with preexisting antenna connectors. In addition to using an external antenna, the antennas on one or both units can be swapped for directional, high gain antennas, if the positions of the units is fixed or known.

To aid with the use of directional antennas, along with increasing the robustness of the system, occasional “heartbeat” transmissions could be made by the transmitter. This way the receiver could tell that the transmitter is within range and that future transmissions can be trusted to not be lost. The use of this functionality could be combined with the use of an adaptive transmission power strategy. In this strategy, the transmitter can begin with a low transmission power and increase it if packets are not received successfully.

Based on the results of the transmitter unit’s battery life tests, the battery of the transmitter could be made significantly smaller, and the specified minimum battery life would still be achieved. This would allow for the protective case to also be smaller, allowing for increased versatility in its placement.

A large improvement in the usability of the system would be to change the motion sensor. The motion sensor was deemed to not fulfill its marketing claims and is not suitable for this application. A large range of PIR sensors are available on the market and a more suitable candidate can surely be found to fit this exact application. Some PIR sensors even come with lower power consumption than the microwave sensor.

REFERENCES

[1] Jeremy S. Cook. The Right Tool for the Job: Active and Passive Infrared Sensors, 11.09.2018. [Internet] Cited: 22.09.2023, Available: https://www.arrow.com/en/research-and-events/articles/understanding-active-and-passive-infrared-sensors

[2] Margush, Timothy S. Some Assembly Required : Assembly Language Programming with the AVR Microcontroller, CRC Press, 2011, Chapter 1, 643 p. [Textbook, Ebook] Cited 23.10.2023, Available: https://doi.org/10.1201/b11791

[3] Labcenter Electronics. Microcontroller Peripherals, 2023. [Internet] Cited 26.10.2023, Available: https://www.labcenter.com/blog/sim-microcontroller-peripherals/

[4] Texas Instruments. Principles of Data Acquisition and Conversion, 2015. [Internet] Cited 26.10.2023, Available: https://www.ti.com/lit/an/sbaa051a/sbaa051a.pdf

[5] Walt Kester. Data Conversion Handbook, Newnes, 2005, Chapter 3, 953 p. [Textbook, Ebook] Cited 26.10.2023 Available: https://www.analog.com/en/resources/technical-books/data-conversion-handbook.html

[6] Geoffrey Brown. Discovering the STM32 Microcontroller, 5.6.2015. [Internet] Cited 26.10.2023, Available: https://legacy.cs.indiana.edu/~geobrown/book.pdf

[7] Stratify Labs. What's a Microcontroller, 2016 [Internet] Cited 20.11.2023, Available: https://blog.stratifylabs.dev/device/2016-12-20-Whats-a-Microcontroller/

[8] Chuck Moozakis, Sean Michael Kerner. What is wireless communications? Everything you need to know, 2023. [Internet] Cited: 26.10.2023, Available: https://www.techtarget.com/searchmobilecomputing/definition/wireless

[9] STMicroelectronics. STLQ020 datasheet, 2021. [Internet] Cited 18.09.2023, Available: https://www.st.com/resource/en/datasheet/stlq020.pdf

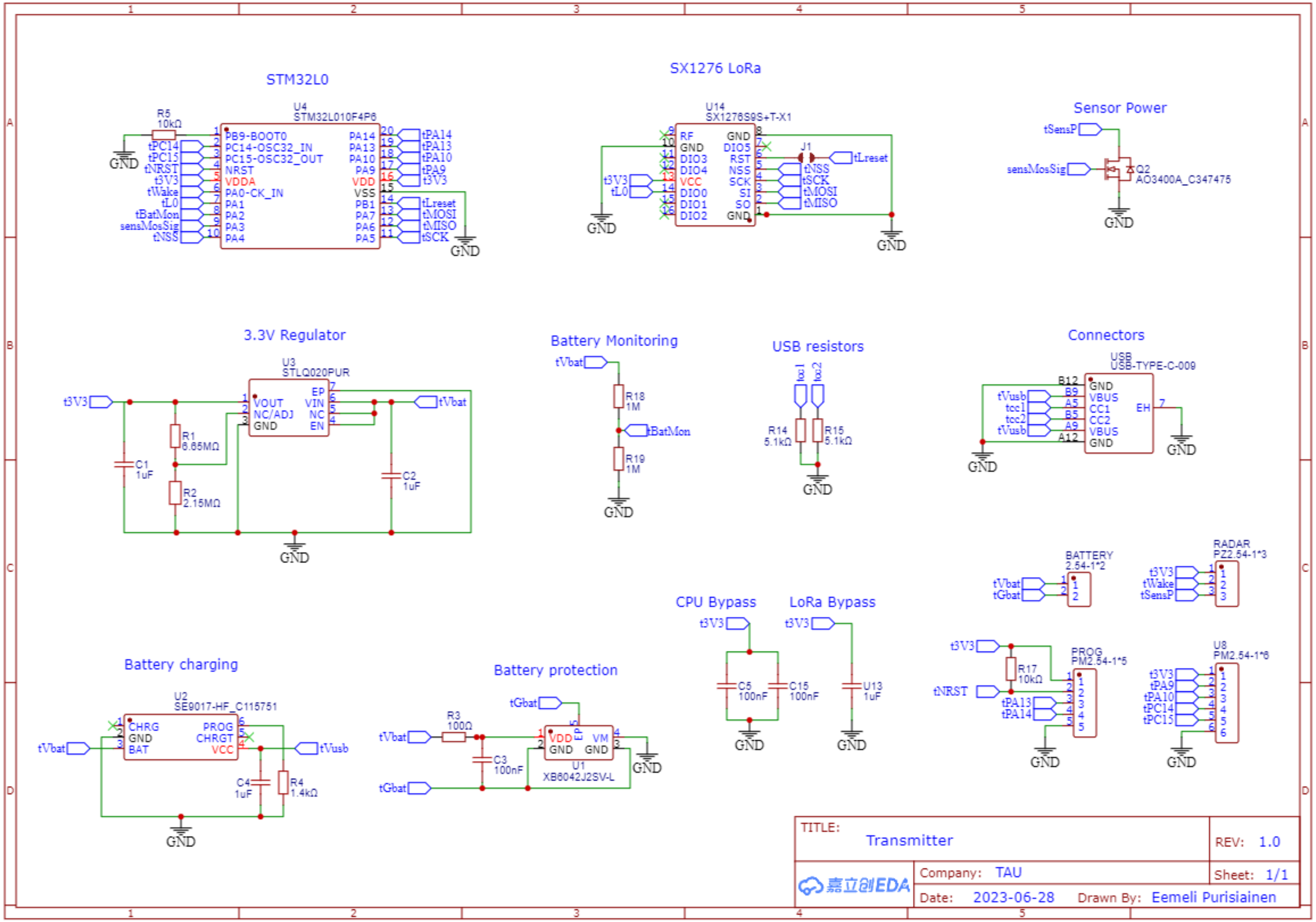
[10] Atmel. ATmega32U4 datasheet, 2016, Page 258. [Internet] Cited 17.09.2023 Available: https://ww1.microchip.com/downloads/en/devicedoc/atmel-7766-8-bit-avr-atmega16u4-32u4_datasheet.pdf

[11] ECC. ERC Recommendation 70-03, 12.02.2021, Page 8. [Internet] Cited 17.09.2023 Available: https://docdb.cept.org/download/25c41779-cd6e/Rec7003e.pdf

[12] Hitex. The Insider's Guide To The STM32 ARM Based Microcontroller, Hitex (UK) Ltd, 2009, Chapter 1, 102 p. [Textbook, Ebook] Cited 17.09.2023, Available: https://www.hitex.com/fileadmin/documents/tools/dev_tools/dt_protected/insiders-guides/stm32/isg-stm32-v18d-scr.pdf

[13] STMicroelectronics. STM32L010F4 datasheet, 2019. [Internet] Cited 17.09.2023, Available: https://www.st.com/resource/en/datasheet/stm32l010f4.pdf

APPENDIX 1: TRANSMITTER UNIT CIRCUIT SCHEMATIC



APPENDIX 2: RECEIVER UNIT CIRCUIT SCHEMATIC