Innovative Livestock Monitoring: Design and Implementation of a GPS-Based Anti-Theft System for South African Farmers

Henry Louw
23837217

Report submitted in partial fulfilment of the requirements of the module Project(E) 488 for the degree Baccalaureus in Engineering in the Department of Electrical and Electronic Engineering at Stellenbosch University.

Supervisor: Dr JM.Strauss

June 10, 2024

UNIVERSITEIT·STELLENBOSCH·UNIVERSITY
jou kennisvennoot • your knowledge partner

Plagiaatverklaring / *Plagiarism Declaration*

1. Plagiaat is die oorneem en gebruik van die idees, materiaal en ander intellektuele eiendom van ander persone asof dit jou eie werk is.
   Plagiarism is the use of ideas, material and other intellectual property of another's work and to present is as my own.

2. Ek erken dat die pleeg van plagiaat 'n strafbare oortreding is aangesien dit 'n vorm van diefstal is.
   I agree that plagiarism is a punishable offence because it constitutes theft.

3. Ek verstaan ook dat direkte vertalings plagiaat is.
   I also understand that direct translations are plagiarism.

4. Dienooreenkomstig is alle aanhalings en bydraes vanuit enige bron (ingesluit die internet) volledig verwys (erken). Ek erken dat die woordelikse aanhaal van teks sonder aanhalingstekens (selfs al word die bron volledig erken) plagiaat is.
   Accordingly all quotations and contributions from any source whatsoever (including the internet) have been cited fully. I understand that the reproduction of text without quotation marks (even when the source is cited) is plagiarism

5. Ek verklaar dat die werk in hierdie skryfstuk vervat, behalwe waar anders aangedui, my eie oorspronklike werk is en dat ek dit nie vantevore in die geheel of gedeeltelik ingehandig het vir bepunting in hierdie module/werkstuk of 'n ander module/werkstuk nie.
   I declare that the work contained in this assignment, except where otherwise stated, is my original work and that I have not previously (in its entirety or in part) submitted it for grading in this module/assignment or another module/assignment.

23837217 Studente nommer/ *Student number*	Handtekening / *Signature*
HC.Louw Voorletters en van / *Initials and surname*	June 10, 2024 Datum / *Date*

Contents

List of Figures

List of Tables

Nomenclature

Variables and functions

R2	Resistance number 2
R1	Resistance number 1
P	Power
V	Voltage
I	current
V_{drop}	Voltage drop
f_c	center frequency

Acronyms and abbreviations

mA	milliampere
uA	microampere
mV	millivolt
V	Volt
t	Time
GPS	Global Positioning System
LoRa	Long Range
LoRaWAN	Long Range Wide Area Network
RX	Receive
TX	transmit
UART	Universal Asynchronous Receiver-Transmitter
SPI	Serial Peripheral Interface
RF	Radio Frequency
MSB	Most Significant Bit
Hz	Hertz
RW	Read/Write
MS	Master/Slave
BDU	Block Data Update
WIFI	Wireless Fidelity
LNA	Low Noise Amplifier
PA-BOOST	Power Amplifier Boost
RFOP	Radio Frequency Operating Parameter
MISO	Master In Slave Out
MOSI	Master Out Slave In
SCK	Serial Clock
IC	Integrated Circuit
Li	Lithium
P-MOS	P-channel Metal-Oxide-Semiconductor
USBASP	Universal Serial Bus Asynchronous Serial Programming
SAPD	South African Police Department
MEMS	Micro-Electro-Mechanical Systems

Chapter 1

Introduction

1.0.1. Background of the project

South African farmers face a grave challenge every day: waking up to the realization that most of their livestock has been stolen or has wandered off. A case study in 2021 demonstrated that over 26300 and 14000 cattle were stolen between July and September in 2020. In Gauteng alone, this issue is costing farmers R1 million a month. Farmers have lost confidence in the SAPS and need an alternative solution to keep track of their livestock in a cheap and effective way. We believe that we can provide an alternative solution by designing a GPS tracker that provides farmers with the necessary information to ensure their livestock is safe and in the designated area . We face several challenges in our design including battery life, signal strength and cost. When designing our GPS tracker we will have to keep all these factors in mind to ensure an effective and cost-efficient GPS tracker.

1.0.2. Problem statement

Stock theft is not a new crime in South Africa and affects millions of farmers every year.The severity of farm attacks and livestock theft has worsened and is not only affecting the livelihood of many farmers, but are also affecting the economy. During the 2011/2012 financial year, SAPS reported that stock theft experienced in South Africa increased by 1.5%. These stock theft crimes are not isolated and relates to more serious crimes, which increased by 20.5%.The recovery rate of stolen sheep dropped from 25% in 2015/2016 to 20%. We believe that many farmers might not report small losses to the SAPS due to its low recovery rate, leading to the increased rate of unreported stock theft cases by 67,7% in 2015 to 70% in 2016. In 2016 a total of R180 086 000 worth of sheep were stolen with only R36 006 000 recovered. This is the enormous problem South African stock farmers are faced with every year.

1.0.3. Objectives and Methodology

We aim to create a GPS tracker that analyzes and studies the behavior of sheep to ensure the safety of farmers' livestock. To do this we are faced with three major challenges as mentioned in our introduction, which is battery life, poor signal strength and cost. Our objective is to design a GPS tracker with low power consumption and long-range communication. To preserve battery life, our GPS tracker will transmit data to the farmer at convenient time intervals. The tracker need not be placed on every sheep, but since they flock together the behaviour of the flock can be interpreted by the behaviour of the sheep wearing the tracker. This information will include the coordinates of the sheep and their acceleration. Using this data we can alert the farmer if their sheep are in potential danger. The farmer will specify the coordinates within which they want their livestock to roam. If the livestock goes beyond the specified coordinates, the GPS tracker will notify the farmer. To maximise the battery life, the device will enter sleep mode until an abnormal movement is observed or until the device has reached a set time interval. We will be using a LoRaWAN(Long Range Wide Area Network) to ensure long range communication. This device is very suitable for our application. The LoRaWAN provides coverage of large geographical areas with low power consumption. For this design, we will focus on a transmitter, that will be attached to a sheep. This device must meet all the specifications mentioned in this chapter.

Chapter 2

GPS module

This chapter describes the function of GPS modules and how they work. Choosing the right GPS module is essential for this application

2.1. What is a GPS and how does it work?

A GPS(Global Positioning system) is a navigation system that works with satellites orbiting the Earth. These satellites orbiting the earth communicate with the receiver by sending signals. These signals contain information about the satellites' location at a specific time. These satellites travel about 20000km above the Earth's surface at a speed of 14000km/h, [12].A GPS needs to communicate with only three satellites to determine its position at a specific time. A fourth satellite is often used to confirm the information sent by the other three.
GPS use a technique called trilateration. Trilateration is a technique that receives signals from at least three satellites. With these signals the GPS can calculate the distance from the satellite to its location by using equation 2.1. Equation 2.1 calculates the distance by multiplying the speed of light by the time it takes the signal to travel from the satellite to the GPS device.

$$Distance = (3.8 \times 10^8) \times t \tag{2.1}$$

The GPS then uses the trilateration method to calculate its position by determining the distance from each satellite in a three-dimensional space.The position of each satellite can be represented by $(x1, y1, z1)$,$(x2, y2, z2)$,and $(x3, y3, z3)$.The r is the distance calculated in equation 2.1 from each satellite. The coordinates x, y, z of the GPS are calculated by using equation 2.2,2.3,and 2.4.

$$r1 = \sqrt{(x - x1)^2 + (y - y1)^2 + (z - z1)^2} \tag{2.2}$$

$$r2 = \sqrt{(x - x2)^2 + (y - y2)^2 + (z - z2)^2} \tag{2.3}$$

$$r3 = \sqrt{(x - x3)^2 + (y - y3)^2 + (z - z3)^2} \tag{2.4}$$

Several factors might affect a GPS module's performance. One of these factors is the number of satellites visible to the GPS. This will have a positive effect on the GPS module by making it more accurate in determining its exact location. Another factor relates to the atomic clock of the GPS. This clock needs to be accurate to calculate precise distances and locations. The atomic clock keeps the time within three nanoseconds. These atomic clocks are essential for GPS applications. The satellites must send their data signals at an exact time. If the GPS's atomic clock is not synchronized, it won't correctly calculate the distance between the GPS and the satellite and, therefore, won't be able to determine its true position. The sensitivity of the GPS is also a factor. Higher sensitivity in a GPS is better, because it allows the GPS to interpret weaker signals or work in harsher conditions.

2.2. GPS hardware connection and software

The GPS module we'll be using is the YIC31616GMSGG GPS module with an integrated antenna. The characteristics of the GPS module will be listed in table 2.1.

The GPS module will be communicating with the micro-controller via UART(TX and the

Parameter	min	Typ.	Max.	Units
Input Voltage	2.8	3.3	4.2	V
tracking	-	29	-	mA
Standby Mode	-	2	-	mA

Table 2.1: Characteristics of a YIC31616GMSGG, [7]

RX pin).A capacitor with the value of 100pF will be connected parallel to the TX and RX pins to cancel out any noise. To limit the current of the signals send via UART a resistor value of 1K will be placed in series with the TXD and RXD pins of the YIC31616GMSGG. The YIC31616GMSGG RXD pin will be connected to the TXD pin of the ATmega328pb-AU micro-controller and the TXD pin will be connected to the RXD pin of the microcontroller.

2.2.1. GPS off when ATmega328PB-AU is off

We want our system to indicate the sheep's location at specific time intervals. For this we will only need the GPS and the micro-controller to be on during these specific time intervals to save power. In our design, we want the GPS to switch off whenever the micro-controller enters sleep mode. For this application, we will use a P-MOS circuit design. P-MOS is an electrical component that has three pins namely: Gate, source and drain. The main power source, 3.3V, will be connected to the Source pin, while the Drain pin will be connected to the GPS module. The Gate pin is the pin that determines whether the current flows from the source to the drain and to the GPS module. When the voltage provided to the gate pin is higher than the Gate threshold voltage, the current will be allowed to flow from the Source pin to the Drain pin. We will use the micro-controller, ATmega328PB, to control the current flowing to the GPS module. We will use one of the micro-controller pins as an input to the gate of the P-MOS to indicate whether current should be allowed to flow through the P-MOS. When the ATmega328PB is sleeping, it won't provide any power to the gate or to the P-MOS, and therefore we will be controlling the power to the GPS module. Consequently, when the ATmega328PB module is sleeping, so is the GPS module in order to save power. The characteristics of the P-MOS can be seen in table ??. Figure 2.1 will give a clear description

Parameter	Min	Typ	Max	Unit
Drain Voltage	-	-	6	V
Gate threshold voltage	1	1.5	2	V
Drain current	-	-	0.3	A

Table 2.2: Characteristics of the PMOS,ISS17EP06LM

of how the GPS will be wired.

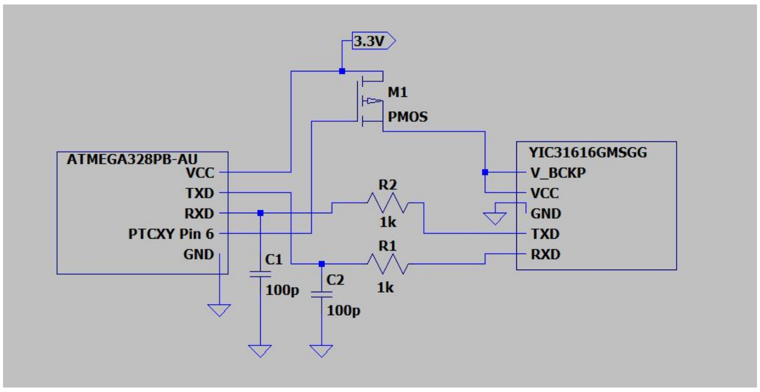


Figure 2.1: Schematic of the GPS connection.

2.2.2. GPS software

When communicating to the GPS with the use of UART we will be using the TinyGPSPlus library by Mikal Hart, [13]. This library allows the initiation of the GPS's TX and RX pins and enables the reading of the serial communications. First we will initiate a condition to check if there is any GPS data available in the serial port. If there is data available to read we will encode the data and check the data validity. If valid the data will be read and stored into a string.

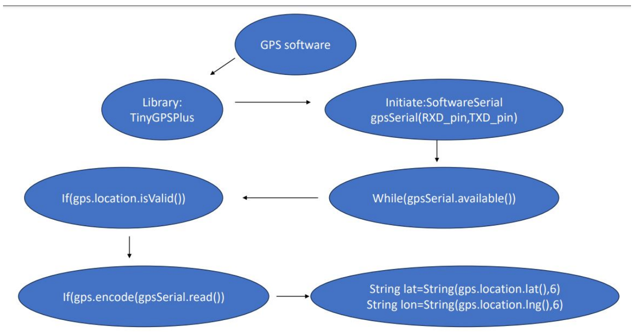


Figure 2.2: Caption

2.3. GPS results

First we need to determine if the GPS is working by determining if the GPS is sending accurate coordinates and by evaluating the TX and RX pins of the GPS. We will also be evaluating the coordinates received and check if they are valid. The coordinates is valid when more than 10 characters are received by the GPS over a span of 5 seconds. Figure 2.3 shows the activity of RX and TX pin of the GPS. This figure shows activity on the RX pin, but not activity on the TX pin. This is as expected. TX has no activity due to our GPS being indoors. The GPS is unable to function correctly indoors, therefore no data is being transmitted, because data cannot be accessed. Rx shows activity therefore we conclude that the micro-controller is successfully communicating with the GPS. How will we then know if GPS is transmitting data correctly? Due to having a USBASP allowing us to program

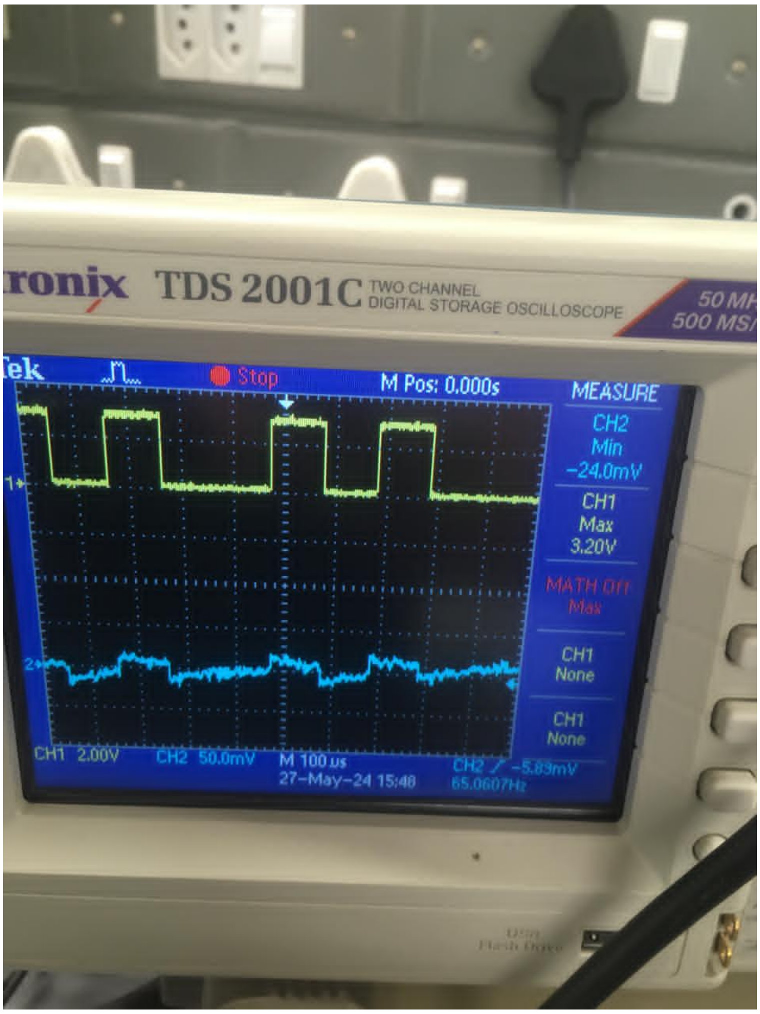
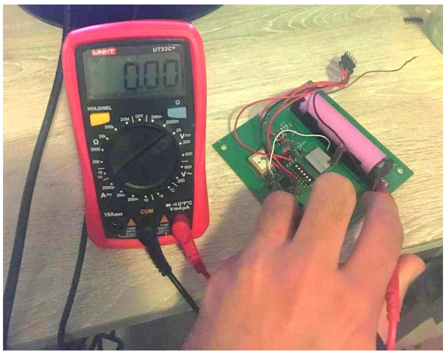
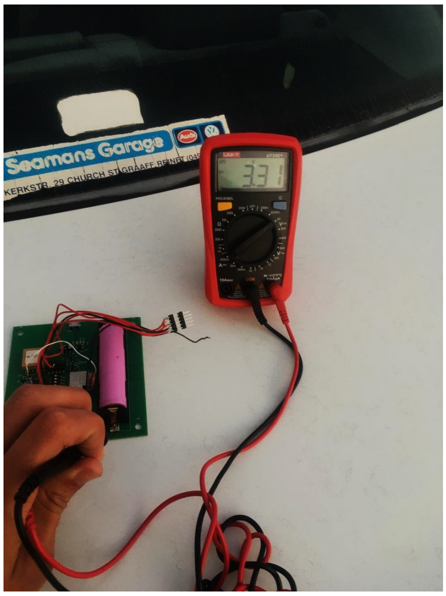


Figure 2.3: CH1(RX) and CH2(TX) pins of UART

and therefore not having any serial connection with an USBASP it will be harder to debug the performance of the GPS. Therefore to debug the performance if the GPS retrieves valid coordinates we will *digitalWrite* pin 7 of the ATmega328PB high if GPS has successfully received and sent the coordinates to micro-controller via UART. These two figure 2.4b and



(a) Pin 7 is Low when GPS is indoors.



(b) Pin 7 is High when GPS is outdoors.

Figure 2.4: The voltage measurement of PIN 7.

2.4a indicates that GPS is transmitting valid coordinates only when the GPS is outdoors as stated by the data sheet of the YIC31616GMSGG. With P-MOS a big mistake was made by believing that current is allowed through the P-MOS when the voltage is higher than the threshold voltage at the gate. The P-MOS actually allows the current to flow from the source to the load when the gate voltage is lower than the threshold voltage. When the voltage is higher than the threshold voltage, it causes the flow of positive charge carriers(holes) from the source to the drain, which stops the current from flowing from the source to the drain. Therefore for the GPS to run we need to *digitalWrite* the pin 6, as can be seen on figure 2.1, low and not high.

Chapter 3

Accelerometers

3.1. Literature survey

3.1.1. How it works

Accelerometers are devices that measure the vibration or acceleration, which is the rate of change in velocity, of a structure, [14]. The accelerometer measures this change in velocity by using MEMS, which are small micro electronic mechanical systems. MEMS are small devices containing silicon springs that bridge two fixed plates that are parallel to one another. When there is a change of velocity it causes the springs to be subjected to force. When the spring is subjected to force, it causes the spring to move, which influences the distance between the two fixed plates. This causes the capacitance between the two fixed plates to change. The change of capacitance generates an electrical signal, which is proportionate to the acceleration. A physical presentation of MEMS is demonstrated in figure 3.1. The acceleration output

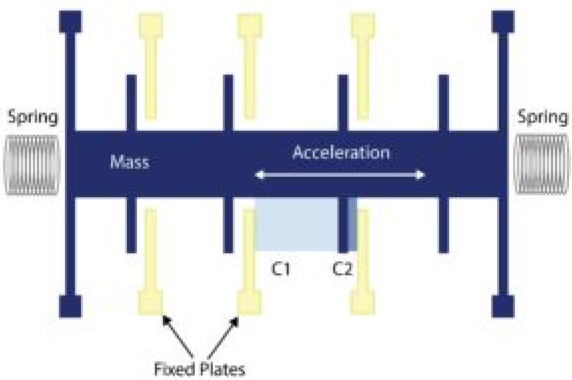


Figure 3.1: a Physical representation of MEMS, [1]

is usually given in "g", gravitational force. The g is a unit based on the acceleration due to gravity at the Earth's surface, which is about $9.8m.s^2$. It is important to consider when working with the z-axis, parallel to the gravitational field, that gravitation will have an effect on your output. When your object is at rest the accelerometer will measure $9.81m.s^2$ in the z direction due to gravitational force. To solve this problem you would need to subtract the gravitational acceleration from your read value to get the true acceleration in the z direction as can be seen on formula 3.1. To calculate the acceleration we take the average acceleration in each axis by using formula 3.2

$$z_{real} = z_{measured} - 1 \tag{3.1}$$

$$Acceleration = \sqrt{x^2 + y^2 + z_{real}^2} \tag{3.2}$$

3.1.2. How it's performance might be influenced

Accelerometers's performances can be influenced. The accuracy of the acceleromters's performance can be influenced by harsh environments like dust or humidity. It is important to choose an accelerometer with a higher sensitivity as that will lead to better efficiency and more accurate reading. An accelerometer also has a frequency range in which it operates. If the accelerations have frequencies out of the accelerometers's range it can cause the accelerometers's accuracy to be compromised.

3.1.3. Objective of the accelerator in our design

The reason to use an accelerometer in our design is to study the behaviour of the sheep. We want to notify the farmer when the sheep are reacting to a threat by running. To do this we will introduce an interruption to the micro-controller every time an unusual or large acceleration takes place. First we need to study the behaviour of sheep and their acceleration during that behaviour.Figure 3.2 will illustrate the amount of acceleration(g) in each of their behaviours to ensure that the information is analyzed correctly. We will use figure 3.2 to code and assess the sheep's behaviour. We will then read the data from the accelerometer and categorize the data based on the information in figure3.2

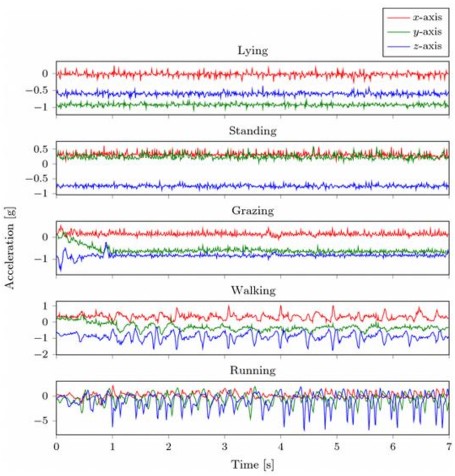


Figure 3.2: Acceleration of sheep at different behaviors at a frequency of 10Hz in 2014. [2]

3.2. software and hardware design

We have chosen to work with the LIS12DE12TR. The LIS12DE12TR characteristics will be tabulated in table 3.1. The LIS12DE12TR is the best application for low power consumption.

	min	typ	max	unit
Voltage supply	1.71	2.5	3.6	V
current consumption normal mode	-	6	-	uA
Sensitivity	15.6	-	187.5	mg/digit
SPI clock frequency	-	-	10	MHz

Table 3.1: Characteristics of the LIS12DE12TR, [8]

The LIS12DE12TR will communicate with the microcontroller, ATmega328PB, with use of SPI communication. The hardware connection between the ATmega328PB micro-controller and the accelerometer, LIS12DE12TR, can be seen in figure 3.3. The capacitors will be used to cancel out all the noise and ensure that the communication between the accelerometer and the micro-controller is effective and reliable.

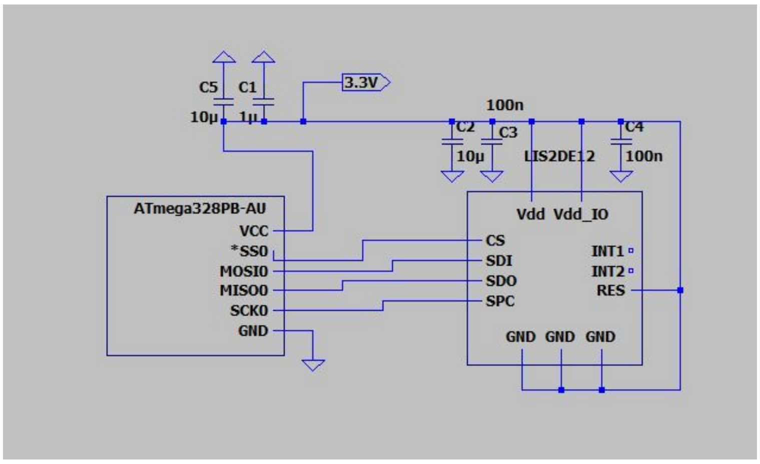


Figure 3.3: The accelermeter's hardware connections.

3.3. Software of accelerometer

We want to communicate to the accelerometer by using SPI communication. To do this we need to look at figure 3.4. To write or to read data from the accelerometer the CS pin needs to be turned to 0 as can be seen in figure 3.4. To do this we digitally write the output pin of the micro-controller to a 0 or a 1 to indicate whether the micro-controller will be reading or writing. When RW pin is 0 it means that the data DI7-DI0 will be written to the address AD5-AD0 with the MSB first. When RW is high it means that the data will be read from D0(7:0) with the MSB first. When the MS is set 0 it means that you will read or write to the same address and the address doesn't get incremented. If the MS is 1 it means that the

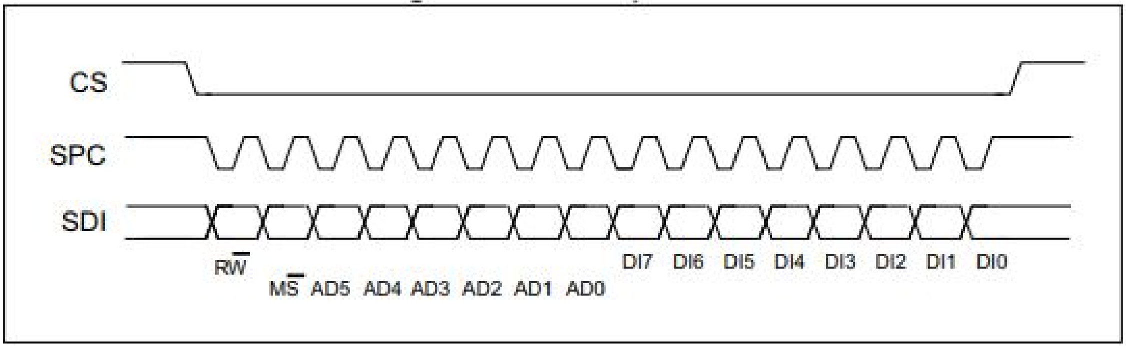


Figure 3.4: SPI-write or read protocol

address will be incremented. When you are writing to the accelerometer we need to consider the data-sheet to ensure that we will be reading and writing to the correct registers.
First we would look at the first applicable register,which can seen on figure 3.5 ODR[3:0]

CTRL_REG1 (20h)

Table 28. CTRL_REG1 register

ODR3	ODR2	ODR1	ODR0	LPen	Zen	Yen	Xen

Table 29. CTRL_REG1 description

ODR[3:0]	Data rate selection. Default value: 0000 (0000: power-down mode; others: refer to *Table 30*)
LPen	This bit must be set to '1' for the correct operation of the device. Default value: 0
Zen	Z-axis enable. Default value: 1 (0: Z-axis disabled; 1: Z-axis enabled)
Yen	Y-axis enable. Default value: 1 (0: Y-axis disabled; 1: Y-axis enabled)
Xen	X-axis enable. Default value: 1 (0: X-axis disabled; 1: X-axis enabled)

Figure 3.5: SPI register that enables the axes and certain functions of accelerometer

is the data rate selection, which would be 50 Hz in our case, which can be written as "0 1 0 0" according to the data sheet. We want all the axes to be enabled along with the LPen to be high for the correct operation of the device. So the data that will be written to $CTRL\_REG1(0x20)$ will be 0x4F.
The next relevant register we can look at can be seen in figure 3.6 We will be doing the same with $CTRL\_REG4$ by using BDU enabled(1) so that there can be no interruption of the data while reading it. The output register would be updated until the full word is read. We will also be working with a sensitivity of +-8g due to the sheep's acceleration as can be seen figure 3.2. The rest would be set to default value. This means we will be writing 0x90 to the the address 0x23.

8.9 CTRL_REG4 (23h)

Table 37. CTRL_REG4 register

BDU	0(1)	FS1	FS0	0(1)	ST1	ST0	SIM

1. This bit must be set to '0' for correct operation of the device.

Table 38. CTRL_REG4 description

BDU	Block data update. Default value: 0 (0: continuous update; 1: output registers not updated until MSB and LSB have been read)
FS[1:0]	Full-scale selection. Default value: 00 (00: ±2*g*; 01: ±4*g*; 10: ±8*g*; 11: ±16*g*)
ST[1:0]	Self-test enable. Default value: 00 (00: self-test disabled; other: see *Table 39*)
SIM	SPI serial interface mode selection. Default value: 0 (0: 4-wire interface; 1: 3-wire interface).

Table 39. Self-test mode configuration

ST1	ST0	Self-test mode
0	0	Normal mode
0	1	Self-test 0
1	0	Self-test 1
1	1	-

Figure 3.6: SPI sensitivity register

Chapter 4

LoraWan

4.1. Technical overview

LoRa is a communication system that involves modulating and demodulating chirp pulses. LoRa encodes information on radio waves using chirp pulses, where the frequency of these chirp signals increase or decrease over a certain time period to avoid interference or noise, [15].The mathematical function of a chirp signal can be seen in equation 4.1.

$$s(t) = cos[(2\pi(f_c t + \frac{\beta}{2}t^2)] \tag{4.1}$$

f_c: is the center frequency, which is the nominal frequency around which the signal frequency varies.

$s(t)$: is the chirp signal

β : is the chirp rate at which the frequency varies.

The LoRa receiver will then demodulate the signal and decode the information provided on the chirp signals based on the frequency rate of the chirp. The LoRa can achieve long distance communication by spreading the signal over a long bandwidth. The signals have binary codes of 0's and 1's that are encoded on the different frequencies varied on the chirp pulses. For example, if a binary code of 0 needs to be transmitted it might use lower frequency than when a binary code of '1' needs to be transmitted. An example of the chirp signal that varies in frequency can be seen on figure4.1. LoRa chirp modulation technique spreads the signals energy over a long bandwidth for a efficient use of energy and it cancels the noise on the signal by using a variety of frequencies. This chirp signal spectrum technique causes Lora to have a long battery life and a long range of communication.
There are many advantages of using LoRa.The first advantage of using LoRa would be that it uses low power and can transmit and receive signals over a long range. It has a low cost with minimal infrastructure and the LoRa module ensures communication that are secure. LoRa can also handle a high capacity of messages from thousands of gateways. LoRa also provides security mechanisms to protect data transmission to ensure confidentiality. LoRa modules also work with unlicensed frequencies bands: 433MHz, 868MHz, 915MHz to avoid interference and legal issues with local spectrum usage.

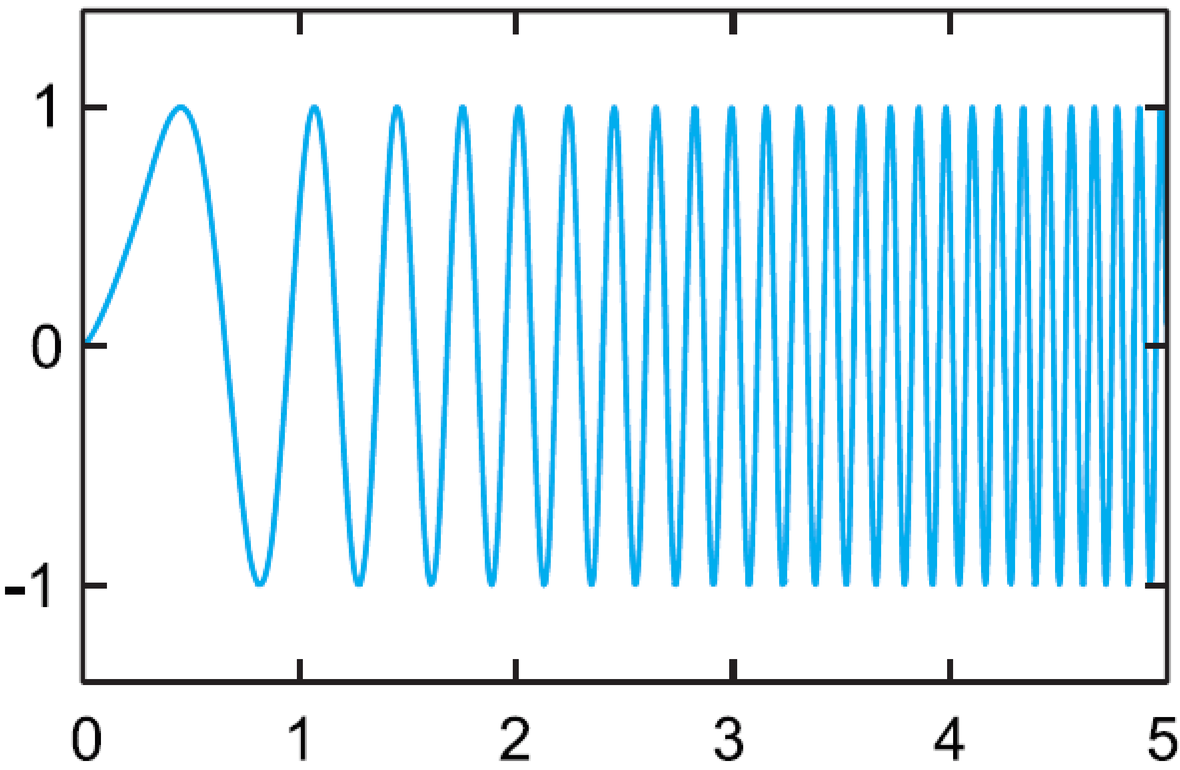


Figure 4.1: LoRa chirp pulse spectrum illustration, [3]

4.1.1. Structure of the LoRa module

The LoRa module exist of a few components: Transceiver, Antenna, Micro-controller, Firmware/Software. The transceiver module is meant for transmitting and receiving signals that are typically radio frequency signals. The Transceivers usually consist of modulated and demodulated signals. The Antenna is also used for transmitting and receiving signals by optimizing communication range. LoRa modules use the micro-controller for data processing and handling of data.

4.1.2. Components of LoRa Communication Ecosystem

LoRa modules receive and transmit the signals over long distance using chirp pulses. The Gateway serve as a bridge between the LoRa module and the LoRaWAN network. The Gateway forward them to LoRaWAN network servers through backhaul connectivity like cellular and WIFI. The LoRaWan will then encrypt and validate the data and send them to an Application server where the information will be assessed, stored and processed. The components of LoRa communication Ecosystem are illustrated in figure 4.2

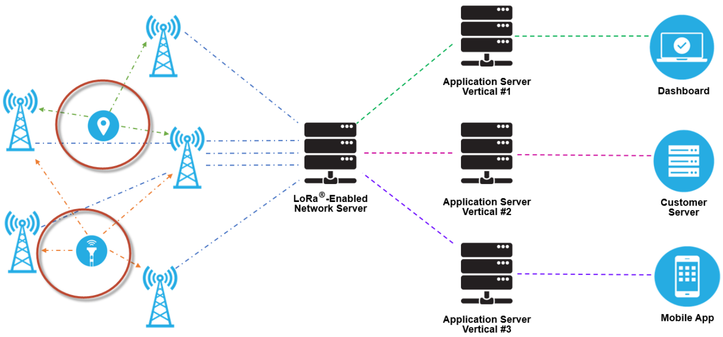


Figure 4.2: Components of LoRa Communication Ecosystem, [4]

4.2. Hardware connection of SXlora1278

We chose the SX1278 LoRa RA-01H to communicate and transmit the information. The maximum ratings and characteristics of LoRa module can be found in table 4.1 We are

Parameter	min	typ	max	units
Voltage supply	1.8	-	3.7	V
Supply current in sleep mode	-	1.5	-	uA
Supply current in standby mode	-	1.6	1.8	mA
Supply current in Transmit mode(13dBm)	-	29	-	mA
Frequency	-	868	-	MHz

Table 4.1: Characteristics of Ra-01H, [9]

going to provide the SxLoRa1278 with a power supply of 3.3V. When the SxLoRa1278 is not transmitting or receiving data we will put the SxLoRa1278 in idle mode to save power. When SxLoRa1278 is in standby mode the LoRa module won't be able to receive or transmit data. To have a successful LoRa communication system we will communicating with a frequency of 868MHz with the use of a transmitter and a receiver. The transmitter will receive a GPS location consisting of longitude and latitude along with the acceleration of the sheep. The LoRa will send the packet of data to the receiver. The receiver will then receive and analyze the data to give the farmer a clear description of the condition of his sheep.

4.3. Software Transmitter

4.3.1. LoRa.begin()

We will be using the LoRa library, [16], but first, we need to understand the content of the library to use it effectively. The Lora device needs to be initialized by setting the LoRa's operating frequency, bandwidth, transmission power, and more. For SPI communication to take place it is important to set the CS pin low to enable SPI communication and to toggle the RESET pin from LOW to high to RESET the LoRa module to ensure that the LoRa module start from a clean state. After that the LoRa would be clear to start SPI communication. First we want to ensure that we are communicating with the correct chip, by reading the REG_Version(0x42), which would typically be 0x12. If this is the case we would initialize the address for RegFifoTxBaseAddr(0x0E), this is the memory base where transmitted information will be stored in memory, by writing 0x00 to the register. We will also be setting up the RegFifoRxBaseAddr(0x0F), this is the point where the data will be written to if an event occurs. These two registers are used to ensure proper data flow of the module. By setting up these two registers the module will know where to start reading the received data and where the transmitted data is stored.

Setting frequency

The LoRa module's operational frequency, also needs to be initialized. We will be working with a frequency of 868MHz, because of our module's capabilities. The formula for setting the frequency can be seen in equation 4.2.

$$Frf(23,0) = \frac{2^{19} \times F_{RF}}{F_{XOSC}} \tag{4.2}$$

F_{XOSC} is the frequency at which the crystal oscillator vibrates when an electric field is applied to it, which is 32MHz. The Frf(23,0), 24 bit, value will be stored in RegFrfMsb(0x06), RegFrfMid(0x07), RegFrfLsb(0x08). The 8 most significant bits (23,16) of Frf(23,0) will be stored RegFrfMsb, the (15,8) bits of Frf(23,0) will be stored in RegFrfMid and the LSB's(7,0) will be stored in RegFrfLsb.

Setting LNA boost

LNA is a low noise amplifier, which amplifies the received information and lowers the noise. This way we can better the quality of our receiver. We are not to concerned about the power usage of the receiver, since the receiver will not be in the field, therefore we will use LNA boost to improve the quality of the receiver. We will therefore write 0x03 to the RegLNA(0x0C) to enable LNA boost as can be seen in figure 4.3.

RegLna (0x0C)	7-5	LnaGain	rw	0x01	LNA gain setting: 000 → reserved 001 → G1 = highest gain 010 → G2 = highest gain – 6 dB 011 → G3 = highest gain – 12 dB 100 → G4 = highest gain – 24 dB 101 → G5 = highest gain – 36 dB 110 → G6 = highest gain – 48 dB 111 → reserved Note: Reading this address always returns the current LNA gain (which may be different from what had been previously selected if AGC is enabled.
	4-3	LnaBoostLf	rw	0x00	Low Frequency (RFI_LF) LNA current adjustment 00 → Default LNA current Other → Reserved
	2	reserved	rw	0x00	reserved
	1-0	LnaBoostHf	rw	0x00	High Frequency (RFI_HF) LNA current adjustment 00 → Default LNA current 11 → Boost on, 150% LNA current

Figure 4.3: Settings for LNA

PA-BOOST settings

PA-boost setting enable the output power of the transmitted signal, which increases the range in which the LoRa module can transmit the data. We want to increase the range of transmission, but also want to save power and therefore we will write the value of 14dBm to RegPaConfig(0x09). The current consumption of each transmit mode can be seen in table 4.2.

RFOP = +20 dBm, on PA_BOOST	120mA
RFOP = +17 dBm, on PA_BOOST	87mA
RFOP = +13 dBm, on RFO_LF/HF pin	29mA
RFOP = +7 dBm, on RFO_LF/HF pin	20mA

Table 4.2: Current consumption of the different transmit modes

4.3.2. LoRa.beginpacket() function

This is a function in the LoRa library that sends the parameters for the transmission that includes the transmission frequency, bandwidth and the coding rate. RegModemConfig1 is the register in which you will write the specifications needed for a successful transmission. The default parameters are already written into the register and therefore we don't want to change it, but rather just want to add an explicit header mode to add flexibility. The explicit header allows a packet with an uncertain length to be sent to the receiver, while the implicit header only allows a constant packet length to be sent. Therefore we will be using the explicit header, since we are uncertain of the packet length we will be sending every cycle.

4.3.3. LoRa.print() function in library.h

With the LoRa.print() function we will be writing our data to the FIFO register for transmission. First, we take our data and set it into 8 bits and send the address of the 8 bits by writing it into the FIFO register. We need to set up the FIFO register by initializing the RegFifoTxBaseAddr, RegFifoAddrPtr by writing 0x00 to both of them. The regFifotxBaseAddr is the register where the base address of the data that will be transmitted is stored. By writing 0x00 to the data it ensures that the module will start writing the data from a known point. RegFifoAddrPtr(0x0D) is set to define the pointer to the base address where the data is being stored in RegFifotxbaseAddr to ensure that FIFO start from the base address set. The data will then be read into FIFO. After this the Reg_mode register(0x01), that controls the operational mode of the module, must be written into TX mode by writing 0x03 to it. After that the data will be ready for transmission.

4.3.4. LoRa.endPacket()

This function sets the mode of the module to transmission mode and also reads the RegIrqFlags register to indicate the condition of the data transmission. First to set the mode of the module, the Reg_mode register(0x01), that controls the operational mode of the module, must be written into TX mode by writing 0x03 to it. After that the data will be ready for transmission. To check if the data has finished transmitting, the interrupt flag 3 needs to be '1'. Therefore if the the third bit is '1' we will update the RegIrqFlags register by writing 0x08 to indicate that the transmission has completed.

4.3.5. Using the LoRa.h library

The LoRa module will be initialized and will receive a packet from GPS. The Software of the LoRa will consist of a receiver and a transmitter. The LoRa will send the packet consisting of the GPS coordination where it will be received by the receiver and compared to a parameter consisting of coordinates. The farmer will set the parameter for the sheep to wander in. If the sheep leaves the parameter the system will notify him. The software schematic can be seen on figure 4.4.

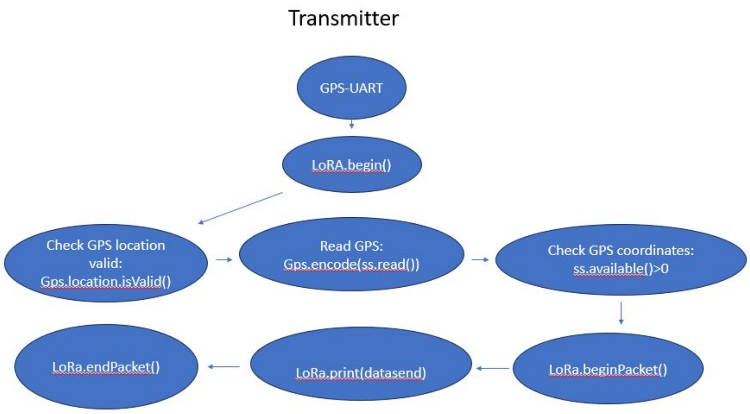


Figure 4.4: Software description of the LoRa transmitter device.

4.4. Software Receiver

For the receiver we will also start by initializing the Lora with the Lora.begin() function that can be described in section in 4.3.1 .Then we will use the LoRa.parsepacket() that we will describe in the next section.

4.4.1. LoRa.parsePacket()

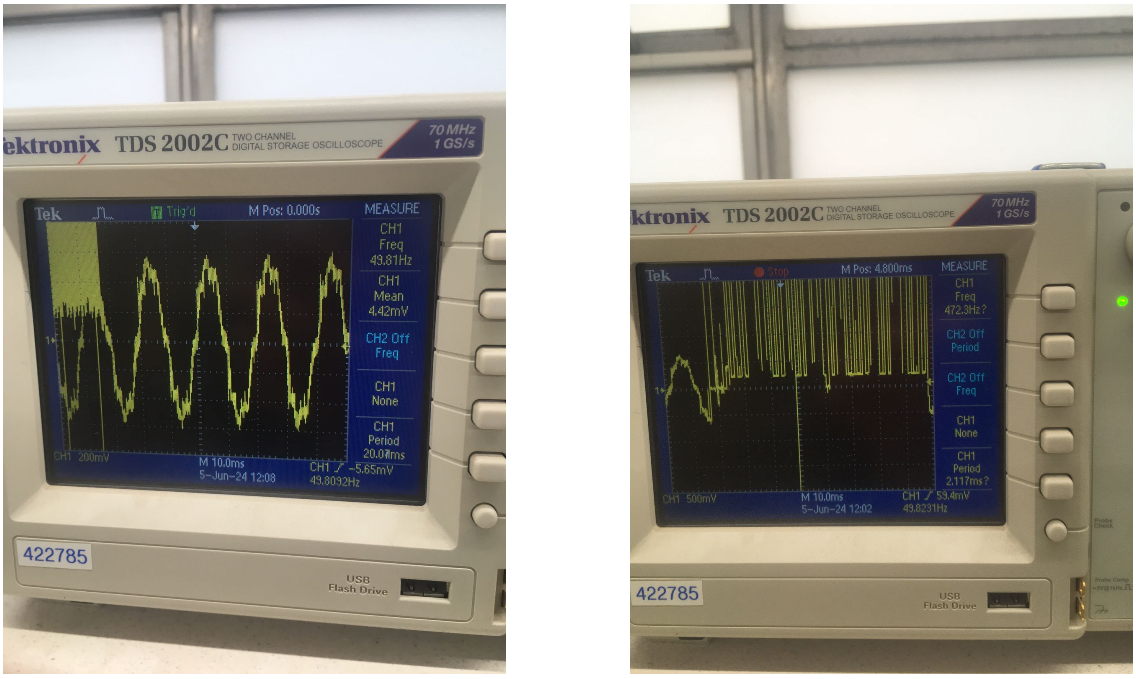
LoRa.parsePacket returns an integer that confirms to the system that a packet has been received. For this to work we need to create a integer *packetlength* in the function. This integer will return the packet length that has been received. To know if we are done receiving the packet we need to check our regIRQ flags system to check if RX is done receiving by the change of the 6th flag in the register. We also need to check if there is any integrity issues with the received data by checking the 5th flag of regIRQ flag register. We will do this by using an 'and' system to check if the 6th flag is '1'(0x40) and the 5th flag is '0'. Then we would know if a packet has been received. If this is the case the packet length will be determined by reading the RegRxNbBytes register. The regRxNbBytes register is used to determine the size of the memory location to be written to during an event or successful transmission.

4.4.2. LoRa.read() and LoRa.available()

When the system received the packet the packet_index was set to 0. The LoRa.available() function reads from the regRxNbBytes to see if there is still bytes to be read by subtracting the packet_index from the read regRXNBBytes value, by increasing packet_index++ after a byte has been read in the RegFifo register with the LoRa.read() function.

4.5. Results

To evaluate the communication between the LoRa and the micro-controller we measure the SPI pins by using an oscilloscope and see if the signals we measure are what we expected. The SPI frequency is set at 200kHz. The results are:



(a) Interference carrier signal of 50Hz **(b)** MISO information on carrier signal

Figure 4.5: Activity of the MISO pin

As can be seen see in figure 4.5 the MISO pin does record activity and the sending of information trying to communicate with LoRa module, but it is doing so on an carrier

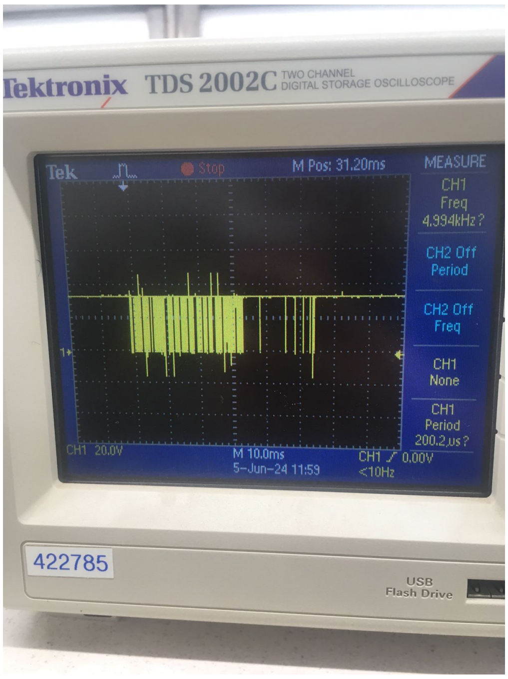


Figure 4.6: Activity on MOSI pin.

interference signal with a frequency of 50Hz. This interference signal can be due to an electromagnetic interference of power lines or power cables near to the MISO pin. To eliminate this a low-pass filter would have to be built to cut off the lower frequency. But other than that the MISO is communicating successfully with the LoRa module and figure4.5 is also indicating that the MOSI pin is sending bits on specific time periods on an high frequency. This is as expected. We set pin 32 and pin 9 on the micro-controller as outputs to indicate whether the LoRa module has successfully been initialized and if the packet has been sent. After some time both were HIGH and therefore we can conclude that transmission of data was successful.

Chapter 5

Battery charger, Python script, Low-voltage protection

5.1. Low voltage protection

Low voltage protection is a mechanism that protects devices or systems when a voltage falls below a certain threshold. The threshold is usually the minimum voltage a system or device can operate at. Low voltage protection prevents system or devices to operate at a voltage lower than their threshold. It is used to prevent systems or devices from being damaged or to operate in an unreliable way. This ensures that the system and devices are functional, reliable and will be able to function for a long duration of time. As the battery is being discharged the voltage will drop, therefore it is important to include low voltage protection in our design to ensure that our components will not be damaged during the discharge of the battery.

5.1.1. Comparator IC

A comparator IC is a integrated circuit, which compares two input voltages and outputs a voltage according to the compared input voltages. In an IC comparator there are two terminals: the positive(+) and the negative terminal(-), the V(+) supply, the V(-) supply, and some IC comparators have a reference pin as well. When the positive terminal of the comparator has a higher voltage than the negative terminal the output of the comparator will be high. If the positive terminal of the comparator is lower than the voltage input at the negative terminal, the output of the comparator will be low. A figure of the comparator and how it would look like can be seen in figure 5.1.

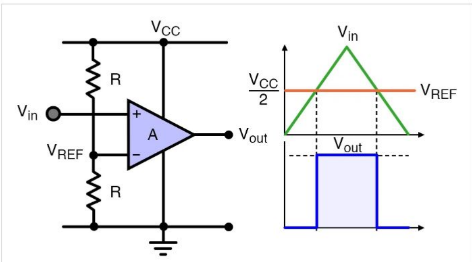


Figure 5.1: The functionality of a comparator, [5]

5.2. Battery charger

The GPS tracker needs a battery that has a long life so that it can stay operational in the field where it can study the behaviour of the sheep. The GPS tracker also needs to be energy effective, rechargeable and adaptable to any conditions. Lithium batteries are the best option for this application. There are two types of batteries that work in this application: Li-PO and the Li-ion battery, [17].

5.2.1. Li-ion battery

Li-ion batteries contain liquid electrolytes that consist of Li salts which dissolves into this liquid. The liquid electrolytes then allows the Li-ions to move from positive to negative electrodes during the use of the battery.

Advantages of Li-ion batteries

Lithium ion batteries are more power efficient and has a large amount of energy relative to their size and weight. Lithium ion batteries have a very reliable performance over many charge and discharge cycles. In the application of the GPS tracker it is important to use a reliable power source, which also has a slow discharge rate and has fast charging capabilities, which in this case makes the Li-ion batteries preferable. Li-ion batteries are more common and also cheaper.

Disadvantage of Li-ion batteries

Li-ion batteries does have some safety concerns in terms of risks when the battery gets damaged or exposed to extreme temperatures, but proper handling and proper usage of the battery would minimize these safety risks and not exclude their use.

5.2.2. Li-polymer battery

The Li-polymer battery contains a gel electrolyte that consist polymer matrix that allows Li-ions to move from positive to negative electrode during discharge, instead of using liquid electrolyte.

Advantage of Li-polymer battery

Li-polymer are more flexible and can be molded into various shapes according to the application of your design. Li-polymer batteries are very applicable for light weighted designs. Li-polymer are safer due to gel like electrolyte and less leakage.

Disadvantages of Li-polymer battery

Li-polymer batteries do have a lower energy density and efficiency than Li-ion batteries, but it's marginally the same. The cost of Li-polymer batteries are higher than Li-ion batteries. Li-polymer batteries do not have such a long cycle life as the Li-ion batteries, which amount to about 300-400 cycles.

When looking at the disadvantages and the advantages of Li-ion and Li-polymer batteries it is clear to see that Li-ion batteries are more applicable for energy efficient and low cost designs, while Li-polymer designs are more applicable for designs that are flexible and light weighted.

5.3. Design of the Low protection voltage

For this design we need a voltage comparator IC with a built in reference voltage to compare the voltage of the battery to. This reference voltage will then determine whether the battery has enough voltage to supply to the rest of the circuit. We chose the TLV3012AIDCKR voltage comparator due to its built-in voltage reference of 1.24V and its power capacity. Table 5.1 contains the relevant information regarding the TLV3012AIDCKR. As can be seen in the

	min	typ	max	unit
Operating voltage	1.8	-	5.5	V
Reference voltage	1.208	1.242	1.276	V
Quiescent current	-	2.8	5	uA
Output current	-	0.5	-	mA

Table 5.1: Characteristics of TLV3012AIDCKR, [10]

table above the TLV301AIDCKR has a very low power requirement and therefore can be used in our design. The figure below describes the structure of our design. The Li battery has

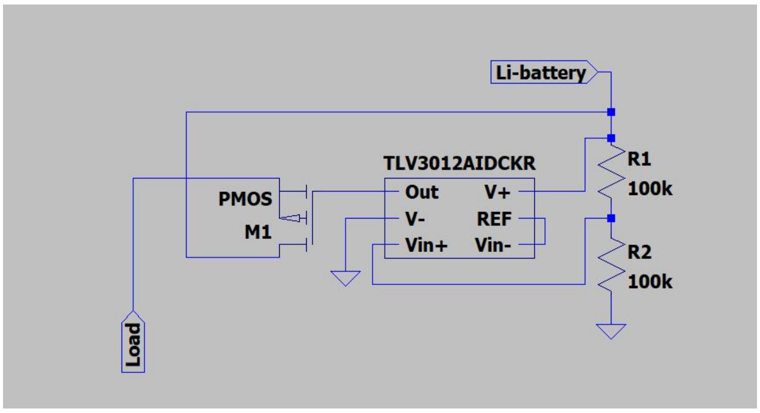


Figure 5.2: Circuit diagram of the low voltage protection

a typical voltage of 4.2V so we want the battery to stop supplying energy to the rest of the circuit when it's beneath 2.484V due to the components of my circuit only operating with a voltage higher than 2.484V. The typical voltage reference of the TLV3012AIDCKR is 1.242V, which means we need to use voltage division to ensure that the comparator provides a low output when the battery voltage is below 2.484V. This signal then needs to be fed into the Positive terminal(Vin+). The calculation of R1 and R2 can be seen in equation 5.1

$$V_{in+} = V_{Li-battery} \times \frac{R1}{R2} \tag{5.1}$$

For this comparator design to work we need V_{in} to be half of what the voltage of the Li-battery is. Therefore when using equation 5.1 R1 needs to be equal to R2. To limit the amount of current flowing through TLV3012AIDCKR we will use two 100kΩ resistors.

5.3.1. Design of the battery Charger

With the battery charger we use a battery charger IC recommended for the charging of Li-battery 18650 cell. We used the battery IC MCP73833-GPI_UN. The MCP is a linear charge management controller that is compatible with potable applications and also provides the battery with a constant current/voltage charge. The specification of the battery charger MCP73833-GPI_UN can be seen in table 5.2. We also used the typical application design for

	min	typ	max	unit
Supply voltage	3.75	-	6	V
Supply current while charging	-	2000u	3000u	A
Regulated output voltage	4.168	4.2	4.232	V
Iout	10mA	-	1A	

Table 5.2: Specifications for the MCP73833-GPI_UN, [6]

our GPS project. This design can be seen in figure 5.3. The MCP73833-GPI_UN charges the

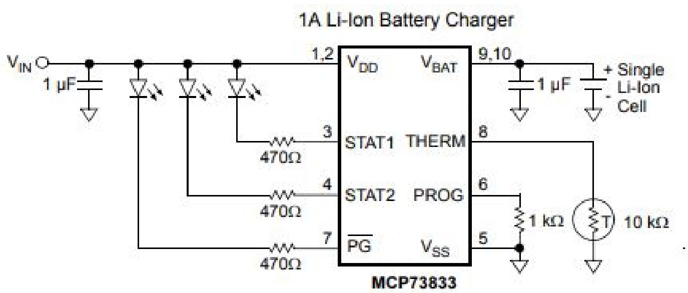


Figure 5.3: Circuit schematic of battery charger, [6]

battery by using a constant voltage and current until it reaches a threshold. When the battery has not yet reached its threshold it stays in a constant current mode to charge the battery quickly. When the battery almost reaches its voltage threshold it lowers its current so that the battery doesn't overcharge. The MCP73833-GPI_UN also has a low current threshold of 50uA to 100uA. When the device approaches this threshold the charger stops supplying current to the battery. The STAT 1 and STAT2 pins are used to indicate the status of the charging process. STAT 1 is used to indicate any fault detection like over-voltage or over-temperature. The STAT 1 pin will pull high when this is the case. STAT 2 is used to indicate that the battery is fully charged or busy charging. The PG pin pulls low when charging is in progress and pulls high when the charging is complete. The LED's are used to display these conditions.

5.4. Python script

We want to develop a Python script that can send a serial message to the receiver, request the received coordinates, and then open the received coordinates in Google maps. The content of the code can be viewed in figure 5.4 and 5.5.

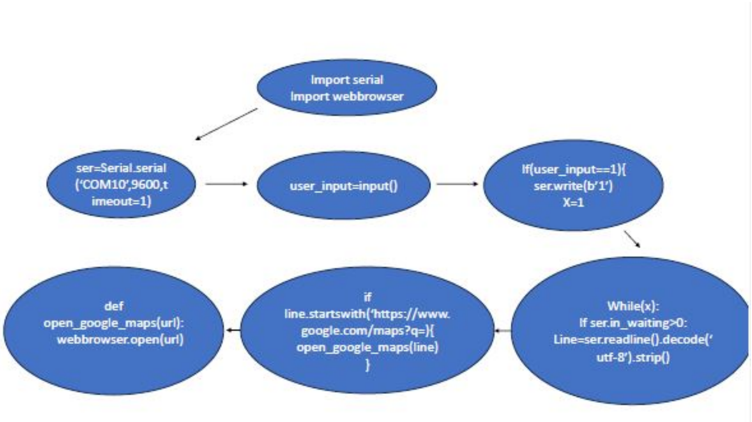


Figure 5.4: Python code

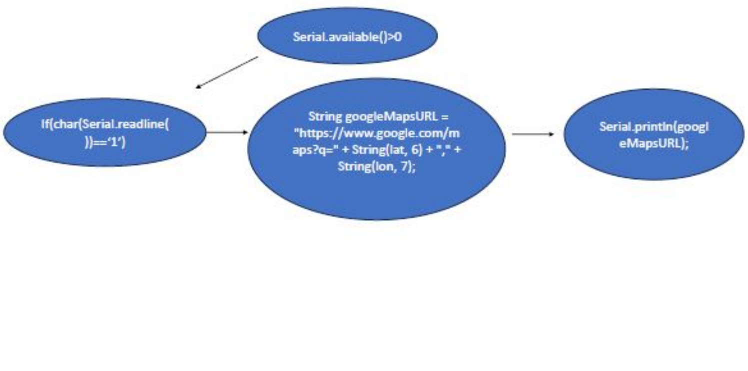
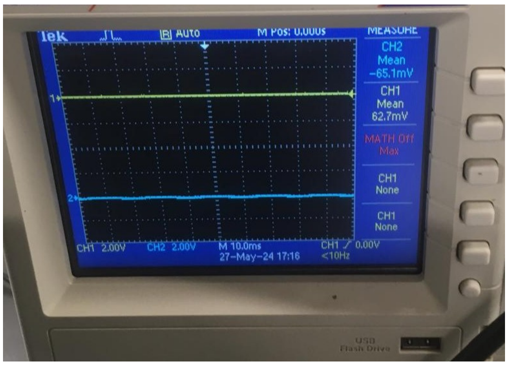


Figure 5.5: Arduino code

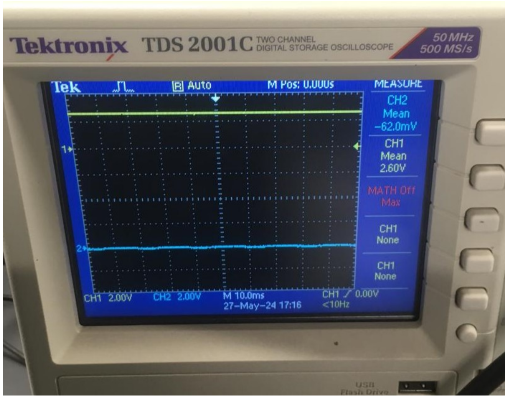
Figure 5.4 is a snippet of the code used in the Python script. The Python script will ask for an input by the user and if the user press '1' then the Python script will write a '1' via serial. The Arduino, figure 5.5, will then check if a command has been received by using the command $serial.available() > 0$. If this is the case the Arduino will write the google maps link and the latitude and longitude coordinates to the Python script via serial. The Python script will then open this link in the web browser.

5.4.1. Results for the Low-voltage protection

To test the low voltage protection circuit we will check the output of the signal by using 2.3V and 2.6V as input to the circuit. We expect the circuit's output to be low when the input voltage is lower than 2.484V and high when the input voltage is higher than 2.484V. To measure the result we used a oscilloscope and a power supply to supply the necessary voltage. The Low voltage protection circuit is producing results as expected. The output of



(a) 2.3V as input.CH1(Output)



(b) 2.6V input-CH1(output) of low-Voltage protection circuit.

Figure 5.6: Results of the low voltage protection circuit

the low voltage protection circuit is low when a voltage lower than 2.484 is used as input to the circuit. As we mentioned previously in the GPS section, we made an error with our understanding of P-MOS circuitry and therefore we had to short out the P-MOS in order for the circuit to work properly.

Chapter 6

System overview and Atmega328PB-AU

In this chapter we will dive deeper into my PCB design and some design decisions, such as using dip switches, component placement and micro-controller selections.

6.1. ATmega328PB-AU

There might have been better options than using the ATmega328PB-AU, but the ATmega328PB-AU micro-controller has the qualities we need to make this project work.

6.1.1. Advantages of ATmega328PB

The ATmega328PB-AU was designed with low power consumption in mind with built-in modes like sleep, idle and power down mode to minimize power usage. The ATmega328PB also has a wide voltage range from 1.8V-5V, which offered flexibility and low power usage. The ATmega328PB also has multiple SPI, I2C, UART and GPIO pins, allowing for greater flexible and functionality. The availability of these pins allowed room for our components and enabled us to have the best system possible. The ATmega328PB also has compatibility with Arduino IDE, which allowes access to different set of libraries. For this reasons we chose the ATmega328PB-AU as our micro-controller. The characteristics of the ATmega328PB is listed further in table 6.1

Parameters	min	typ	max	units
Voltage supply	1.8	-	5	V
Current consumption-Active mode	-	0.24	-	mA
Current power save mode	-	1.3	-	uA

Table 6.1: ATmega328PB power characteristics according to [11]

6.1.2. Programming of ATmega328PB

With a lot of research we came upon the BMT AVR/51SER USBASP, [18] programmer that would allow us to program onto the ATmega328PB via the SPI pins. The Atmega328PB had two SPI pins we could use: the SPI1 and The SPI0 pins. Unfortunately we made the mistake of using the SPI1 pins as programming pins, which will be discussed in chapter 8. Due to the SPI1 pins already being used by the RA01H-LoRa module, we decided to use an 12 pin dip switch. This allowed us to program and send SPI signals to the LoRa module through the same SPI pins. When we were done programming, we turned off the SPI pins connected to the programming pins and put on the LoRa pins via the dip switch. This can also be seen in figure 6.2.

6.2. System overview

In this section we summarise our entire system. The figure of the system can be seen in figure 6.1, 6.2. This system will be used to track sheep's location and to ensure that they are safe. The power supply of our system will be a Li-ion 18650 battery, because of it's energy efficiency and adaptability. For the battery we also built a battery charging system to ensure the longevity of the battery. Due to the battery's voltage range and the characteristics of components of our system we had to use a 3.3V regulator to ensure the functionality and reliability of our system. We also had to build a low voltage circuit to also ensure the functionality and reliability of our system by using an comparator IC, which is explained in Chapter 5. The 3.3V regulator then powered the whole system with 3.3V. The GPS module was also connected to the micro-controller via the UART pins to enable serial communication enabling commands from the micro-controller to the GPS module, but also enabling GPS coordinates from the GPS to the micro-controller. The LoRa module, RA01H, was connected to the micro-controller via the SPI1 pins, which was also connected to the header pins for programming purposes. The programming header pins and LoRa module was separated by using dip switches to prevent interference between radio signals and programming commands. The accelerometer was connected to the micro-controller via the SPI0 pins to measure the acceleration of the sheep. With this system we will be able to transmit the necessary information to notify the farmer if their sheep has left a certain location and are therefor in possible trouble.

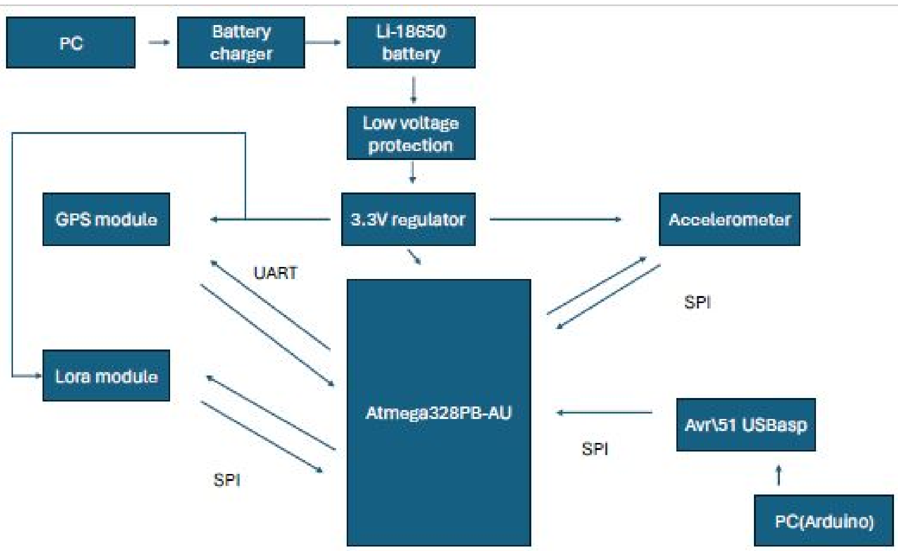


Figure 6.1: Circuit schematic

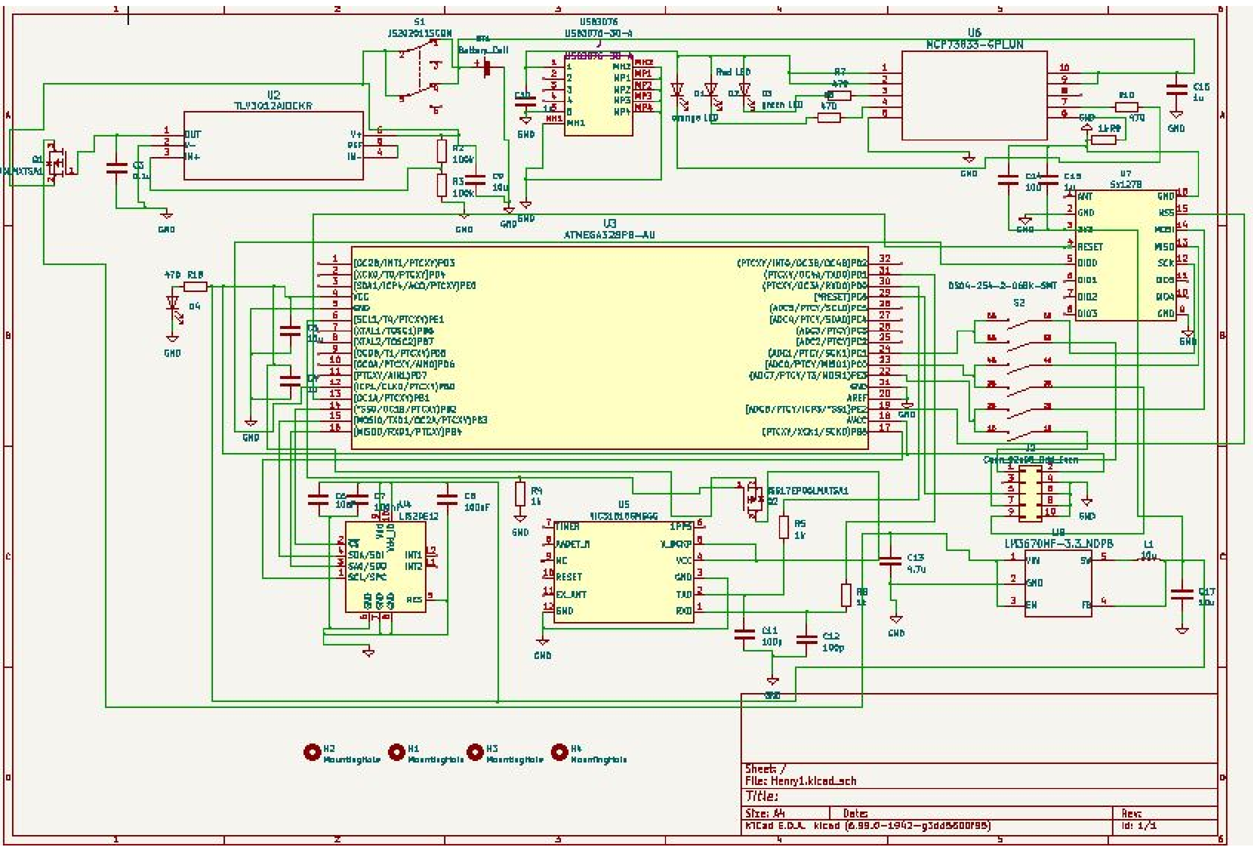


Figure 6.2: Circuit schematic

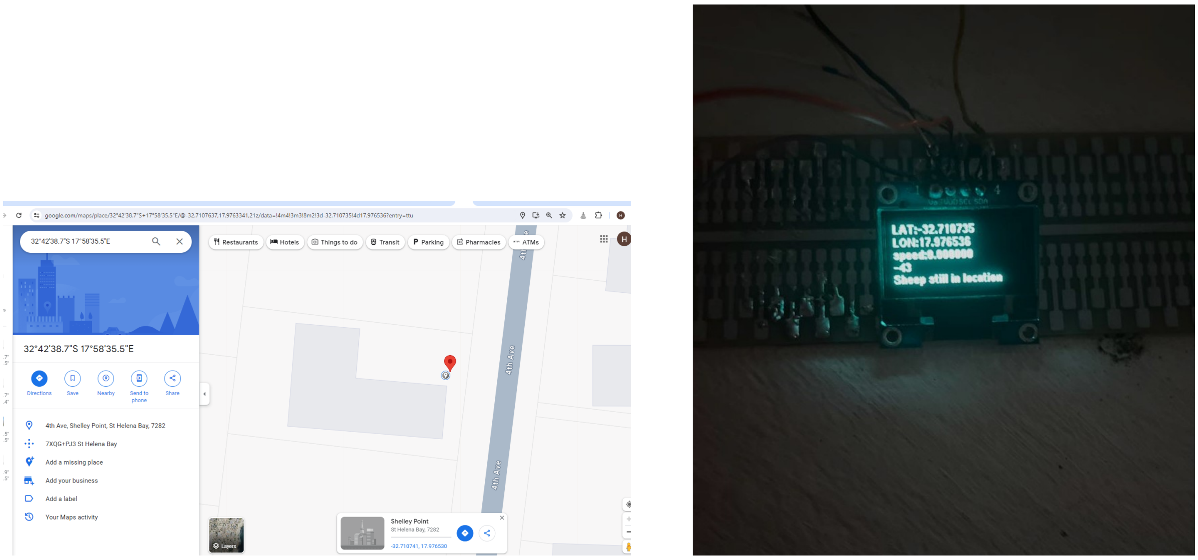
Chapter 7

Overall results and test.

We will conduct several tests to analyze and evaluate the system's performance. For the first test, we will verify whether the LoRa module accurately transmits and receives the correct coordinates.

7.1. True location

It's important to verify whether the LoRa module is initializing and receiving a packet from the transmitter. To determine the true location of the transmitter, we conducted several tests to ensure the accuracy of our system. These tests involves placing the transmitter in known locations and comparing the GPS data received by the LoRa module with the actual data. Sub sequentially, we wrote all the data from the receiver's ESP32S3 to the OLED. We also used the Python script mentioned in Chapter 5 to open a web browser with the transmitted coordinates. When we did this we got the results seen in figure 7.5a. As seen in figure 7.5a



(a) Location GPS device(Red) and my real location(grey) on Google maps..

(b) OLED displaying coordinates,RSSI(signals strength).

Figure 7.1: GPS displaying my true location.

the GPS tracker has an accuracy of about +-2.5m as also stated on the datasheet. Figure 7.5b is where the OLED is displaying all the relevant information. This includes all the data received form the transmitter: longitude, latitude and the speed. The RSSI, the signal strength between transmitter and receiver, is also displayed underneath the speed. Then it also states whether the sheep is still in boundary or not. This boundary was set prior to testing to simulate usage by the farmer. This topic will be expanded upon in the next section.

7.1.1. Boundary Set

We've decided to set the boundary manually by setting four coordinates in an array. We then evaluated whether the coordinates from the sheep, that we received, is within this boundary or not. We want the OLED to display whether the sheep is in or out of the given boundary. The boundary can be viewed in figure 7.2. We conducted a boundary test by asking an individual

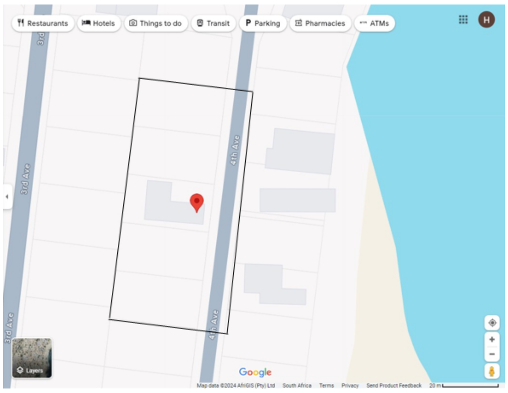
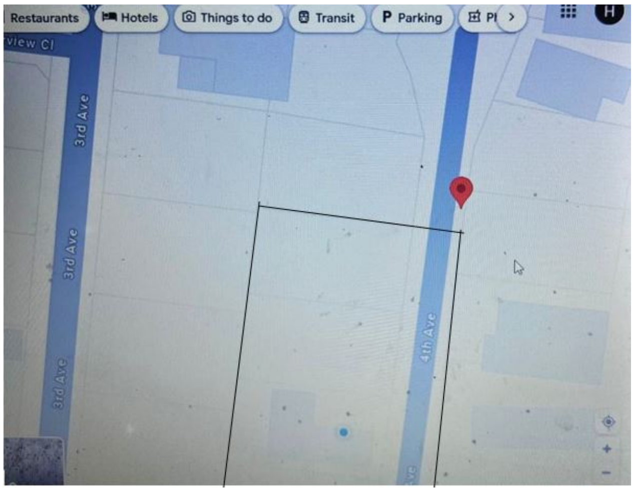
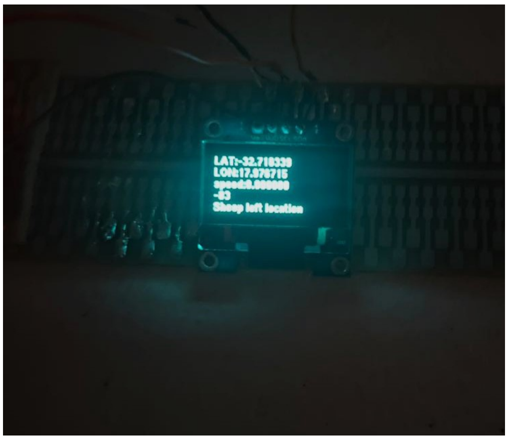


Figure 7.2: The boundary,the black rectangle, set manually.

to carry the transmitter until the OLED displayed that the sheep was out of bounds, while the receiver remained on the porch. On figure 7.3,7.4,7.5 we can see the results. We have three figures: two are out of bounds and one stayed within the boundary.



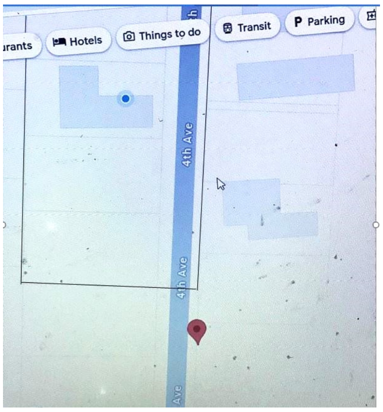
(a) Location GPS device(Red) and receiver location(blue) on Google maps..



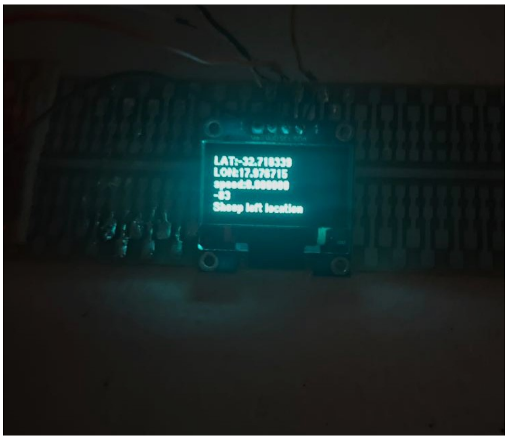
(b) OLED displaying "sheep left location".

Figure 7.3: System demonstrating the boundary test

As can be seen in these figures, the data displayed on the OLED's correspond with the location of the individual. The time it took for the coordinates to be updated took about 5-10s, but my design's antenna was not placed optimally and therefore it took longer for a



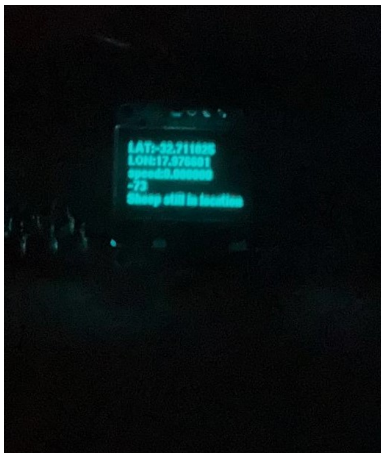
(a) Location GPS device(Red) and receiver location(blue) on Google maps..



(b) OLED displaying "sheep left location".

Figure 7.4: System demonstrating the boundary test.

(a) Location GPS device(Red) and receiver location(blue) on Google maps..



(b) OLED displaying "sheep still in location".

Figure 7.5: System demonstrating the boundary test

packet to be transmitted. We can therefore conclude that the design is a success and truly displays the correct location of the individual.

7.2. Power usage

The power usage is very important in my application therefor it is important to test the power usage of the overall system. We will be using equation 7.2

$$I = V_{drop}/R \tag{7.1}$$

$$P = V \times I \tag{7.2}$$

We can't measure the power over an oscilloscope, because an oscilloscope doesn't measure current. We will connect a known resistor value and connect it in series with the battery powering the circuit. We will then place the multi-meter parallel to resistor to measure the voltage difference over the resistor. With this information we can use 7.1 to calculate the

State of Tracker	V_{drop} over 0.3Ω	$I_{calculated}$
Tracker sleep mode-Gps and LoRa is off	0.4mV	1.333mA
Tracker is on - Gps and LoRa module is functional	1.8mV-3.3mV	6mA-11mA

Table 7.1: Power usage of system

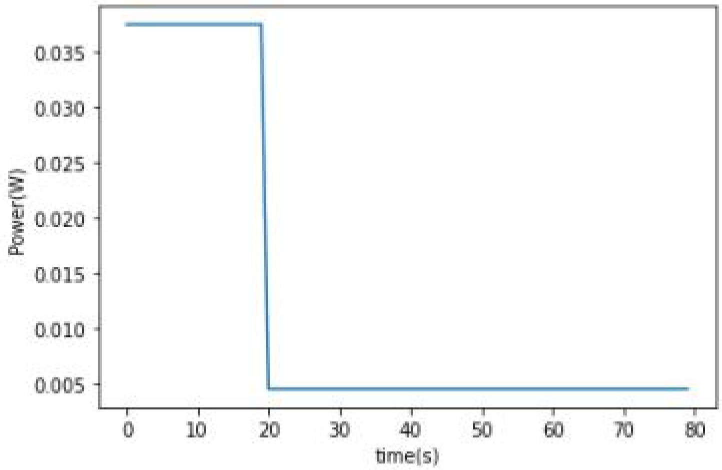


Figure 7.6: Power usage of system per cycle

current and the power flowing through the system. We will be measuring the current when the GPS and LoRa module is in idle and active mode. The LoRa will then transmit the packet and the LoRa module and GPS will then return to sleep mode. When using equation 7.2 we can calculate the power over a period of time. In our case the GPS and LoRa module is on until they successfully transmitted a packet, then they are switched off for 56 seconds. The time it takes for packet to be transmitted is about 10-20 seconds. Therefore your power usage can be view in figure 7.6 over a time span of 80 seconds.

The Li-battery has a battery capacity of 1200mA/h. The time of each cycle is 76 seconds. We can then use equation 7.3,4.7,7.5,7.6 and 7.7 and table 7.1 to calculate the amount of time the system can operate without needing to be recharged.

$$Averagevoltage = \frac{Initial\_voltage + final\_votage}{2}. \tag{7.3}$$

$$Energy\_Capacity(Wh) = Battery\_capacity(Ah) \times Average\_voltage(V). \tag{7.4}$$

.

$$Energy\_consumed = I_{calculated} \times Average\_voltage \times duration. \tag{7.5}$$

.

$$Cycles = \frac{Energy\_Capacity}{Energy\_consumed} \tag{7.6}$$

.

$$Time(sec) = Cycles \times cycle\_time \tag{7.7}$$

. When using the formula we calculated that the system would be able to function for 514 hours without needing to recharge. Therefore we conclude that the system would be able to stay 21days and 10 hours in the field before needing to be charged.

Chapter 8

Conclusion

The designed GPS tracker was successful in communicating the sheep's location and determined if the sheep were within the boundary determined by the farmer. The system's integration of GPS and LoRa module enabled precise location tracking of the sheep and notified farmers that their livestock could be in danger due to theft. The GPS tracker provides a viable solution to South Africa's livestock theft problem and enables farmers to be confident in the safety of their animals.

8.1. Summary

The power efficient 18650 Li-ion battery successfully powered the system with the use of the 3.3V regulator, using a maximum of only 0.0374W of power. This ensures that the battery will be able to be used in the field for the duration of about 21 days, based on results attained in chapter 7. This is sufficient, but not optimal and we would have loved to extend the systems field time. We believe this was due to our error in using P-MOS circuitry, making it impossible to let the ATmega328PB micro-controller go into sleep mode. The ATmega328PB would be unable to shut down the GPS when it is in sleep mode because of the circuit used in figure 2.1. The system's low voltage protection ensured its functionality and reliability. The GPS tracker worked and was able to produce a GPS coordinate accuracy of +-2.5m.

8.2. Recommendation for future work

Even though the GPS tracker successfully communicates the location of the sheep successfully we would also love for the system to indicate the behaviour of the sheep by using data and classifying the sheep's behaviour by using machine learning. We were unable to this due to the SPI0 pins being the true programming pins and therefore we had to remove the accelerometer to ensure successful SPI communication between the PC and the Atmega328PB, because the accelerometer was causing an interference. We would also build bigger antenna's to ensure longer range of communication between the farmer and the sheep.

Bibliography

[1] Level Developments, "MEMS Accelerometer," Available at: https://www.leveldevelopments.com/2020/10/what-are-inclinometers/mems-accelerometer/, accessed: Insert Accessed Date Here.

[2] ResearchGate, "Raw acceleration measurements from the 2014 sheep dataset," Available at: https://www.researchgate.net/figure/Raw-acceleration-measurements-from-the-2014-sheep-dataset-Typical-acceleration_fig3_321183489, accessed: 19 April 2024.

[3] Sghoslya, "LoRa is Chirp Spread Spectrum," Available at: https://www.sghoslya.com/p/lora-is-chirp-spread-spectrum.html, accessed: 2 April 2024.

[4] Altair, "LoRaWAN Network Fundamentals," Available at: https://help.altair.com/altair-iot-studio/topics/edge_ops/lorawan_network_fundamentals.htm, accessed: 3 April 2024.

[5] All About Circuits, "Op-Amp Voltage Comparator Circuit," Available at: https://www.allaboutcircuits.com/video-tutorials/op-amp-voltage-comparator-circuit/, accessed: 15 April 2024.

[6] Microchip Technology Inc., "PIC16F87XA Data Sheet," Available at: https://ww1.microchip.com/downloads/en/DeviceDoc/22005b.pdf, accessed: 16 March 2024.

[7] Yic Company, "XT324-Series," Available at: https://www.yic.com.tw/wp-content/uploads/2023/01/XT324-Series.pdf, accessed: 30 March 2024.

[8] STMicroelectronics, "Datasheet: STM32L071xx, STM32L072xx, STM32L073xx," Available at: https://www.st.com/content/ccc/resource/technical/document/datasheet/group0/72/cb/80/a7/5e/fe/46/9d/DM00153214/files/DM00153214.pdf/jcr:content/translations/en.DM00153214.pdf, accessed: 19 April 2024.

[9] Ozdisan, "Document 632831," Available at: https://cdn.ozdisan.com/ETicaret_Dosya/632831_134737.pdf, accessed: 5 April 2024.

[10] Texas Instruments, "TLV3011: Nanopower Comparator with Voltage Reference," Available at: https://www.ti.com/lit/ds/symlink/tlv3011.pdf?HQS=dis-dk-null-digikeymode-dsf-pf-null-wwe&ts=1717772117376&ref_url=https%253A%252F%252Fwww.ti.com%252Fgeneral%252Fdocs%252Fsuppproductinfo.tsp%253FdistId%253D10%2526gotoUrl%253Dhttps%253A%252F%252Fwww.ti.com%252Flit%252Fgpn%252Ftlv3011, accessed: 12 March 2024.

[11] Microchip Technology Inc., “ATmega328PB,” Available at: https://ww1.microchip.com/downloads/en/DeviceDoc/40001906A.pdf, 2017, accessed: 15 March 2024.

[12] Geotab, “What is GPS?” Available at: https://www.geotab.com/blog/what-is-gps/, accessed: 1 April 2024.

[13] Mikal Hart, “TinyGPS++,” Available at: https://github.com/mikalhart/TinyGPSPlus, accessed: 1 May 2024.

[14] Omega Engineering, “Accelerometers,” Available at: https://www.omega.com/en-us/resources/accelerometers#:~:text=An%20accelerometer%20is%20a%20device,the%20force%20exerted%20upon%20it., accessed: 19 April 2024.

[15] Prajzler, “What is LoRa? The Fundamentals,” Available at: https://medium.com/@prajzler/what-is-lora-the-fundamentals-79a5bb3e6dec, accessed: 2 April 2024.

[16] S. Mistry, “LoRa Library for Arduino,” Available at: https://github.com/sandeepmistry/arduino-LoRa/blob/master/src/LoRa.h, accessed: 10 April 2024.

[17] Robocraze, “Lithium-Ion vs Lithium-Polymer Battery,” Available at: https://robocraze.com/blogs/post/lithium-ion-vs-lithium-polymer-battery#:~:text=Li%20polymer%20battery%20is%20safe,charging%20time%20is%20very%20short., accessed: 7 May 2024.

[18] “PDF Document,” Available at: https://pdf.direnc.net/upload/show.pdf, accessed: 30 April 2024.

Appendix A

Social contract

UNIVERSITEIT·STELLENBOSCH·UNIVERSITY
jou kennisvennoot • your knowledge partner

E-design 344 Social Contract

2023

The purpose of this document is to establish commitment between the student and the organisers of E344. Beyond the commitment made here, it is not binding.

In the months preceeding the term, the lecturer (Thinus Booysen) and a few paid helpers (Jacques Wust, and Michael Ritchie) spent countless hours to prepare for E344 to ensure that you get your money's worth, that you are enabled to learn from the module, and demonstrate and be assessed on your skills. We commit to prepare the assignments, to set the assessments fairly, to be reasonably available, and to provide feedback and support as best and fast we can. We will work hard to give you the best opportunity to learn from and pass analogue electronic design E344.

I, 23837217 .... have registered for E344 of my own volition with the intention to learn of and be assessed on the principals of analogue electronic design. Despite the potential publication online of supplementary videos on specific topics, I acknowledge that I am expected to attend the scheduled lectures to make the most of these appointments and learning opportunities. Moreover, I realise I am expected to spend the additional requisite number of hours on E344 as specified in the yearbook.

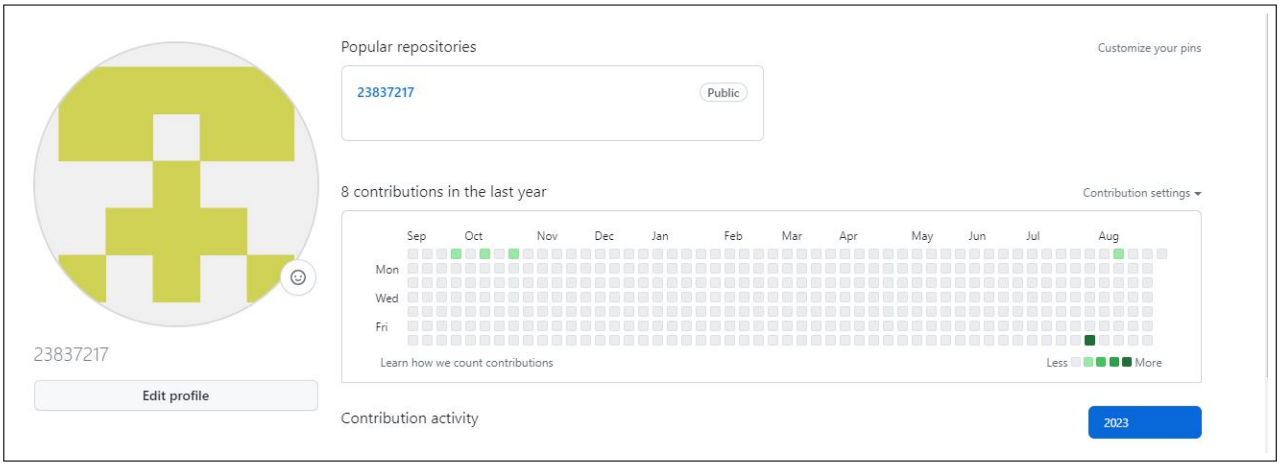
I acknowledge that E344 is an important part of my journey to becoming a professional engineer, and that my conduct should be reflective thereof. This includes doing and submitting my own work, working hard, starting on time, and assimilating as much information as possible. It also includes showing respect towards the University's equipment, staff, and their time.

Prof. MJ (Thinus) Booysen	Student number: 23837217
Signature: MJ Booysen (Digitally signed by MJ Booysen Date: 2023.07.22)	Signature: Henry Louw (Digitally signed by Henry Louw Date: 2023.07.30)
Date: 22 Jul 2023	Date: 30 Jul 2023

1

Appendix B

GitHub Activity Heatmap



Appendix C

Stuff you want to include