Ministry Of Higher Education

and Scientific Research

Final Year Project Report

Sector: Computer engineering

License in: Embedded Systems and Internet of things

Directed by: Ahmed Zgolli

Supervised by:

Academic Supervisor: Doctor Mahmoud Ammar

Professional Supervisor: Jecer Ben Hammouda

Subject: Data acquisition via i2c and transmission via LoRa

College Year: 2023/2024

Dedications

After giving thanks to almighty God I Dedicate this work to:

To my dear Parents who have always been there for me throughout my journey in the university and always showing love and support until the end of my internship I hope they see my gratitude through this dedication and work.

To my dear Brothers and sisters although they live far away but they never fail to make me happy and show me their support and care throughout my internship. I am grateful for them.

To my late grandmother (RIP) who is always praying for my success and always hoped to see me graduate no amount of words can describe how much I miss her.

To all my friends to all those I love who have been supportive and there when I needed them and cheer me up whenever I feel down.

Ahmed Zgolli

Thanks

First, I would like to express my deep gratitude to Be Wireless Solutions for giving me the opportunity to do my final internship within their organization. I wish to express my gratitude to my educational institution, Central University, especially the Director of the Institute and the Head of the IT Department, for allowing me to carry out this internship and acquire concrete training in my field of study. I am grateful for their support and commitment to my professional success. I would also like to thank my teaching assistant, Dr. Mahmoud Ammar, and the university professors in the IT department at Central University in Tunis for the exceptional quality of their supervision, assistance, and presence throughout the development of this project. Additionally, I thank my professional supervisor, Mr. Jecer Ben Hammouda, for his guidance, support, and valuable advice throughout my internship. His expertise and experience have allowed me to acquire important skills and develop my self-confidence.

I also thank all the members of BWS, especially the members of the Embedded Engineering team, for their warm welcome, availability, and collaboration. Their kindness and professionalism allowed me to feel comfortable and quickly integrate into their work environment.

Finally, this internship has been a rewarding and unforgettable experience for me, and I am grateful for all the people who helped me succeed in this important stage of my life.

Summary

Figures List

Tables List

Abbreviations List

- BWS = Be Wireless Solutions
- IoT = Internet Of Things
- I2C = Inter Integrated Circuit
- LoRa = Long Range
- LoRaWAN = Long Range Wide Area Network
- MCU = Microcontroller Unit
- UML = Unified Modeling Language
- UART = Universal Asynchronous Receiver-Transmitter
- IAR = ngenjörsfirman Anders Rundgren

General Introduction

It is in the context of the above perspectives that the idea of smart cities of the new age has developed into a powerful tool for change with the help of information technology in order to improve the quality of life in cities and their efficient and eco-friendly management. With this transition towards integration and information-based decisions, various advancements in technology have been seen across several technological industries. Leading this charge is the Internet of Things (IoT), which has a crucial role to play in advancing smart city concepts. The IoT environment is all about the massive organization of interconnected objects, devices, sensors, and systems that capture, process, and share data, which in turn proactively generate actions that can optimize the manner cities function as a whole. These applications range from smart transport and energy consumption, smart waste disposal and management, smart safety and security, among others, are transforming the operations of cities and their interface with the residents. This work can be aptly classified under the IoT innovation domain since it aims at coming up with an anti-theft system for sensors in BWS, a company that is into the technological advancement business. Utilizing the Internet of Things, this project is to improve safety and decrease theft rates and, at the same time, optimize processes in BWS. In this report, we discuss the designation and deployment of such a system in more detail, including technological options, issues encountered, and mitigating measures that were adopted. This project also effectively demonstrates the practical application of IoT in solving existing problems and further emphasizes the enormous possibilities of using IoT as a key to the creation of the cities of the future.

Chapter 1: The Host Company And General framework

I. Introduction:

The commencement of this chapter sees us delving into a detailed exploration of project, which we undertake. It covers its overall purpose and the underlying intent. The main consideration for this paper is the company that will be hosting the program; where the project will occur. The company in industrial scale this project will be described with details about the company's operation and why the project is crucial to the industry will be presented to give the audience a better understanding of the reason for the project and how it correlates to the industry context. What follows is the explanation of the project's framework, which is the core of the whole formulation and defines the structural elements and the organizational components of the journey. We will map the key elements and strategies that will form the basis of our activities and therefore provide us with the integral system for further analysis and evaluation of the expended work. Thus, the expository introduction will create the solid platform on which the whole book will be built outlining the key elements and indicating what will be discussed in the detailed chapters follow. This chapter acts as a base for a detailed study of the project. It starts by giving an overview of the project, which includes both the theoretical model and the practical experiment.

II. Host organization:

1. Introduction:

Figure 1: Be Wireless Solutions company Logo

This graduation project was carried out within the Tunisian start-up Be Wireless BWS-solutions. Be Wireless Solutions (BWS) was created in 2016, after have obtained the IoT license in 2017 and start-up label in 2019. BWS is the leader in Tunisia in the development of IoT solutions for the economy of resources, including:

- **Wireless sensors**: for data collection and remote control of connected objects.
- **A connectivity offer:** installation and operation of an IoT network (LoRa).
- **Business applications**: web and mobile to monitor connected objects in real time and remotely. BWS provides integrated and connected systems to save resources in various fields: industry, agriculture, energy, transport and smart cities.

2. BWS solutions:

BWS develops solutions that allow real-time monitoring and distance from connected objects: fridges, machines, water meters, vehicles, green and agricultural spaces (henhouse, greenhouse, beehive, hutch), and even a city to save resources.

Among the products offered by BWS, we can cite the best known and requested, namely:

- **Cold chain monitoring solution**: A cold chain monitoring solution for your goods, whether stored or in transit, in real time, remotely, personalized, configurable and wireless. You are then instantly alerted and can know exactly when and where the cold chain has been broken
- **Intelligent Transport Solution**: A customizable, integrated and easy-to-use solution for vehicle tracking.
- **Energy management solution:** A solution for optimizing energy management by instant monitoring and remote reading of energy consumption and detection of equipment energy guzzlers.
- **Water consumption monitoring solution**: A solution to optimize water consumption thanks to real-time remote reading of water consumption and detection of leaks and fraud. The solution applied to mechanical counters without modifications or additional investments

- **“Smart cities” solution** : The “smart cities” solution offers you solutions to put technology back at the heart of your smart city and improve the daily lives of all, through :
  - "Useful" smart services for its citizens.
  - User-friendly apps for city officials and citizens.
  - Better city information sharing.
  - Permanent IoT connectivity and outdoor Wifi offered to the whole city.
- **Smart farming solution :** The smart farming solution allows you to manage your entire space remotely and in real time agriculture thanks to :
- The management of temperature, humidity and brightness of agricultural areas (henhouses, greenhouses, beehives and hutches).
- Soil moisture monitoring and irrigation management (automatic or manual). BWS solutions are wireless, connected, scalable, customizable, easy to install, and above all economical.

3. BWS products:

BWS products, illustrated in the figure (2), are totally Tunisian and produced locally. These are wireless products, easy to install and economical. BWS has a wide range of products:

- Sensors for remote control of connected objects.
- Sensors for collecting data from connected objects.
- IoT gateways for the distribution of the LoRa network allowing the transfer of data from the sensor to the server to access the application afterwards.

Figure 2: BWS products

III. Project framework:

1. Problematic:

A rise in theft and unauthorized access to electronic devices like sensors in Tunisia has raised concerns. Many businesses face security risks and financial losses due to inadequate anti-theft systems. Some key challenges include:

- Protecting devices from theft and misuse remains a significant challenge. Current security measures fail to fully cater to Tunisia's unique requirements, leaving electronics susceptible to unauthorized access and loss.
- Inadequate systems against intrusion pose a significant challenge. Without robust mechanisms for real time surveillance and alerts, unauthorized access or theft incidents may go unnoticed, heightening the peril of data compromise and ope-rational disruptions.
- Anti-theft systems can pose difficulties when merging with current security measures. Compatibility clashes may arise, potentially weakening the system's overall protection.

2. Project objectives:

The project aims to develop a smart antitheft system that is:

- Efficient.
- Stable.
- Low power consumption.
- Reliable.

3. Existing state analysis:

Now, the anti-theft plan within BWS the company is only but a theory, with no practical implementation whatsoever. Proposed project aims to explore a new method for theft and losing of opt-in electronic hardware. The project embarked upon was this having to do with pressing issues as to manage them efficiently, for the time being there is only a LoRa device that is capable of sending data of sensors without anything for antitheft. The present theoretical solution that sounds great on paper, however, has not been deployed, meaning that there is no effective security system in place for the business. This project is the first concrete step in the right direction – the final goal of which is the installation of an anti-theft system, which will be backed up by sensors and an Internet of Things approach. Such project pursues not merely the objective of making BWS Company more secure by protecting its assets but also provides an impulse for innovation and further development in the field of asset security. This strategic approach is in line with the company's mission to be the leading entity in the provision of security and asset management that meets the highest standards of excellence and improvement hence becoming a leader.

IV. Conclusion:

Ultimately, this chapter has given a comprehensive account of our project host, BWS, and this chapter has set the broader purposes of our project. A thorough explanation of the terminology and devices serving as the core of our anti-theft system will take place as we go to the next chapter. Moreover, a detailed needs analysis, which is all about the analysis and comparison of each equipment, will be done, and it will greatly aid us in our selection of the appropriate accessories. In the next Chapter, we start discussing the project concept, the basis for its implementation. The fundamental knowledge that will be discussed in the implementation stage will also come to bear when dealing with detailed technical aspects.

Chapter 2:

Conception

I. Introduction :

This chapter talks about the technological framework, project conceptualization and therefore anti-theft system solutions. We will initially give a broad overview of the technological choices, which include hardware and software selections that are most critical for the development. We continue this by describing the project's vision, mission, and principles it was based on. This involves the UML diagrams for system modeling and visualization we mentioned. The solution implementation hereby, we are going to focus on a holistic system blueprint that will merge perfectly with the strategic goals of the system. This plan encompasses all the later phases of design and development and, in this way, all the component of the antitheft system are integrated into one system and to reach out the best results.

II. Proposed solution :

Security for electronic devices is extremely important nowadays. A solution for our previous problematic that covers all aspects is essential to reduce the growing risks of theft and unauthorized access, making the overall security stronger. The proposed solution is an anti-theft measures involve using smart sensors for continuous tracking. If someone attempts to steal, a warning signal immediately notifies you. This IoT based solution greatly improves security monitoring with minimal hassle.

III. Technological choices :

1. Internet of things:

1.1 What is Internet of things? :

The Internet of Things (IoT) refers to a network of physical devices, vehicles, appliances, and other physical objects that are embedded with sensors, software, and network connectivity, allowing them to collect and share data. IoT objects, like ‘smart devices’ can range from the simplest ‘smart house’ things as smart thermostats to a refined clothing as wearables, to industry machines with RFID tags, and even to transportation systems like smart cars. Technologists had this vision of smart cities that are based on the tools of IoT technology advancements. IoT allows smart devices to interact with each other and with other printable gadgets that are stay over the internet, thus procuring tremendous networks of devices that have the ability of communicating and performing calculations autonomously. Examples are monitoring environmental conditions like in farms, managing smart vehicles with traffic routes, managing machines in factories, and tracking inventory order/shipment in warehouse.

1.2 Internet of things features:

Our anti-theft system is fully based on the Internet of Things (IoT) to introduce many useful features and connectivity functions into our solution. IoT features are integrated seamlessly into our system to enable a wide range of capabilities:

- **Real-time Monitoring:** Our system is equipped with IoT, which is used to have access to the real-time status of connected devices and sensors. Through this functionality, possible aberration or unsuitable actions in the system can be seen off immediately thus make the system safe.
- **Remote Access and Control**: With IoT, users are able to manage our system from wherever location using a smartphone or PC. With this feature screen users will experience comfort and convenience to handling with security settings on the move.
- **Data Analysis and Insights:** We are provided with the ability to collect and analyze data from different sources using the Internet of Things (IoT); this helps to generate constructive information about how devices are being used and the security trends. Being data driven enables preemptive security options and upgrading system efficacy.
- **Scalability and Flexibility:** Our IoT solution can easily adapt its architecture to fit scaled up or down depending on the dynamically changing needs. The latest generation of devices

as well as new sensors can be connected with the system and guarantee capability of adaptation as well as future proofing.

- **Interoperability:** To achieve this, our anti-theft system has been purposely created to be compatible with other IoT devices and platforms allowing for smooth communication through the power of integration. The interoperability capability will foster setting up a comprehensive ecosystem of security.
- **Energy Efficiency:** The Internet of Things in our system is made such that there is energy efficiency to ensure that the features are performing and consuming energy at the low level. On the other hand, using eco-friendly methods not only slashes opportunities costs but also brings about environmental sustainability.

By leveraging these IoT features, our anti-theft system provides a comprehensive and advanced security solution that is not only effective but also adaptable to the evolving needs of our users.

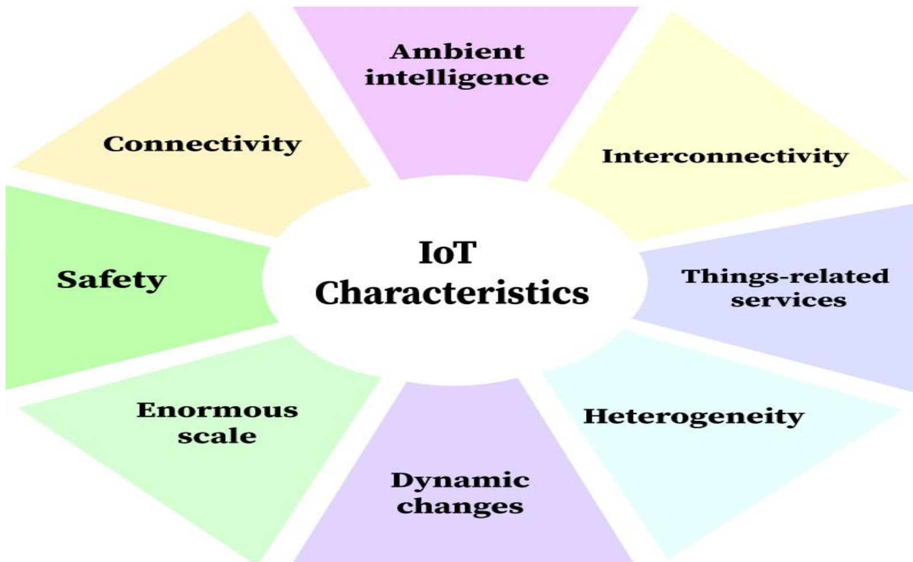

Figure 3: IoT Characteristics

1.3 Internet of things architecture:

IoT architecture is the structure that allows for interconnected devices, cloud services and protocols to create an IoT ecosystem. This network consists of smart sensors, actuators and other connected elements that enable data flow from physical sources through networks into storage in the cloud. The primary purpose of this type of architecture is managing information collected by IoT-enabled technologies so it can be analyzed or processed accordingly. Devices, sensors &

network infrastructure all need to work together if a successful communication between them needs to take place. Once done they are capable of fulfilling certain tasks such as troubleshooting any issue discovered within the system thanks to efficient data analysis techniques deployed via its components like smart devices wireless sensors.

Its essential role remains constant collecting user provided/observed attributes plus providing feedback on their effectiveness during operations; while also controlling processes with precise measurements made possible due native capabilities

Figure 4: IoT architecture

- **Application Layer:** In IoT, the application layer is where specific applications and services are executed that use IoT devices and machine learning methods. It is involved with people and other means; it includes interfaces for computers representation of information, analytics, and devices control.
- **Enablers Layer:** The enablers layer in IoT provides the foundational technologies and solutions that

enable IoT functionality. This includes protocols, standards, and middleware that facilitate data exchange between connected devices and manage IoT networks. This layer ensures interoperability, scalability, and security in IoT systems.

- **Platforms Layer:** Platforms in IoT are software frameworks that offer services and tools for developing and managing IoT applications and devices. They provide features like data and device management, connectivity, security, and application development tools. IoT platforms simplify development, aid integration with diverse devices, and enable scalable and secure IoT solutions.
- **Protocols Layer:** The protocols layer in IoT consists of communication protocols that govern how devices and systems exchange data. These protocols ensure compatibility and define rules

for data transmission. Examples include MQTT, CoAP, and HTTP, each designed for specific purposes like efficient data transfer or low power consumption**.**

- **Connectivity Layer:** The connectivity layer in IoT refers to the infrastructure and technologies that enable devices to connect to the internet and communicate with each other. This layer includes technologies such as Wi-Fi, Bluetooth, ZigBee, and cellular networks, which enable devices to establish connections and transmit data.
- **Objects Layer:** The objects layer in IoT consists of physical devices connected to the internet that can collect and exchange data. These devices include sensors, actuators, and smart appliances, enabling real-world data collection and smart functionalities.

2. LoraWAN:

2.1 What is LoRa and LoRaWAN? :

LoRa is a radio modulation technique that manipulates radio waves to encode information using chirped, multi-symbol format. It can refer to the systems supporting this modulation or the IoT communication network. LoRaWAN is a low power, wide area networking protocol built on LoRa. It wirelessly connects devices to the internet and manages communication between end-nodes and network gateways.

2.2 LoRa and LoRaWAN functioning:

2.2.1 LoRa:

LoRa uses a method called spread spectrum. The purpose, however, remains the same: to be able to transmit at the same time, on the same channel. The LoRa protocol uses seven "codes" called Spreading Factors [SF6, SF7, SF8, SF9, SF10, SF11, and SF12], which allow seven simultaneous transmissions on the same channel. The signal emitted by the LoRa modulation is a symbol whose basic shape is represented below. Its name

(Chirp) comes from the fact that this symbol is used in Radar technology.

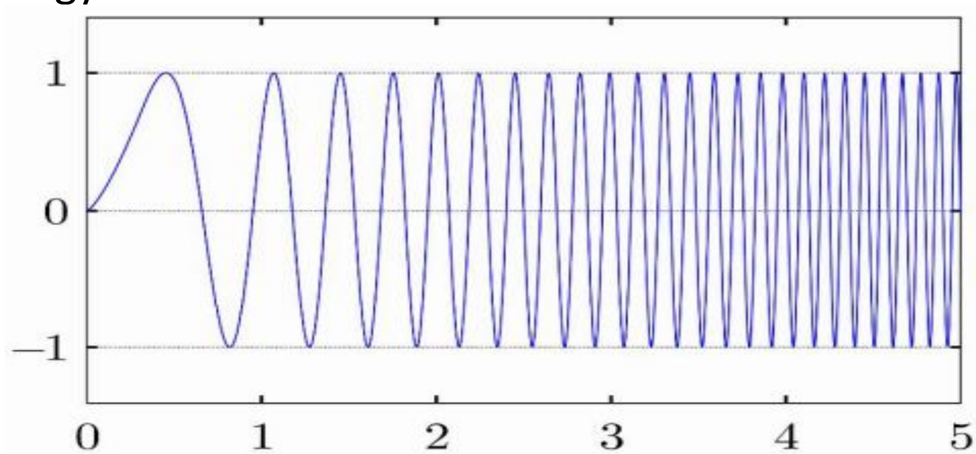

Figure 5: Symbol of the LoRa modulation

The start frequency is the central frequency of the channel minus half the bandwidth and the end frequency is the central frequency of the channel plus half the bandwidth. In conclusion, LoRa modulation offers a robust and efficient solution for long-range communication in IoT applications. By utilizing the Chirp Spread Spectrum modulation scheme, LoRa enables reliable data transmission over extended distances while maintaining low power consumption. This unique modulation technique, coupled with its adaptability to various bandwidths, makes LoRa an ideal choice for IoT applications requiring reliable, long-range connectivity.

2.2.2 LoRaWAN:

LoRa is the technology that enables long-range communication, while LoRaWAN is the protocol that governs how devices communicate over that network. Here we have the architecture of the LoRaWAN that is composed of:

- LoRa Devices: LoRa Devices are physical devices equipped with LoRa technology that transmit and receive data over LoRaWAN networks.
- LoRa Gateways: LoRa Gateways are devices that receive and transmit data between LoRa devices and a LoRaWAN network, serving as the bridge between end-devices and the network server.
- Network Server: A Network Server in LoRaWAN is a central component that manages the communication between LoRa devices and applications. It handles the routing of messages, security, and network configuration.
- Application Server: An Application Server in LoRaWAN is responsible for processing and storing data received from LoRa devices. It handles the application-specific logic, such as

data parsing, storage, and integration with other systems or services.

- User Application: the software or program that runs on the end user's device or system, utilizing data received from LoRa devices via the LoRaWAN network. This application can perform various tasks, such as data analysis, visualization, or control actions based on the received data

Here is a figure that represents the architecture of LoRaWAN:

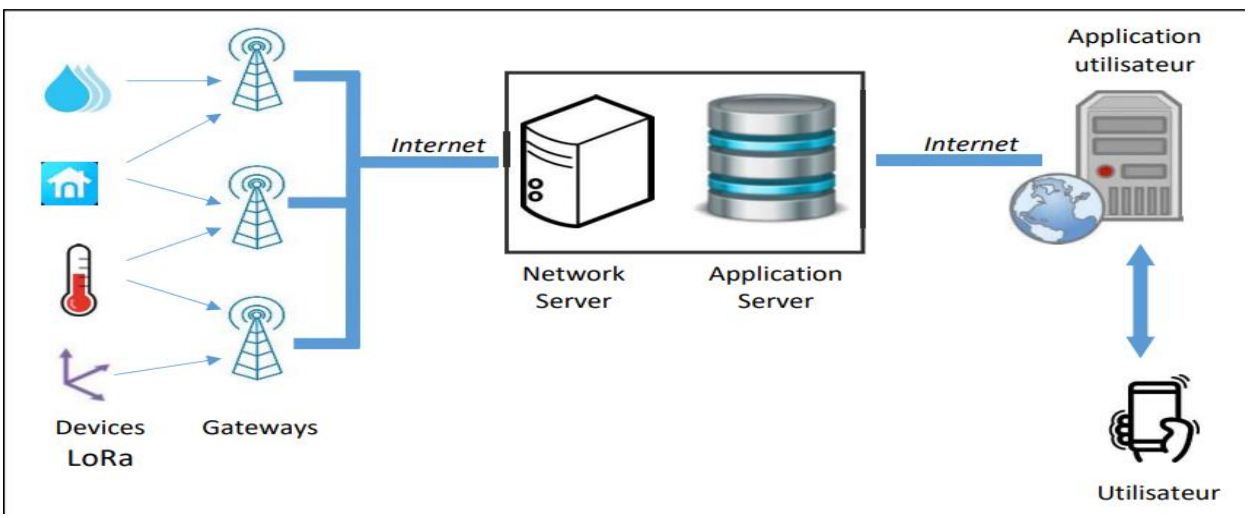

Figure 6: Architecture LoRaWAN

As shown in Figure 5 the components explained earlier all function together and in that order in a LoRaWAN network, which will be similar to the way, our antitheft system functioning when it comes to sending data.

2.3 LoRa packet structure:

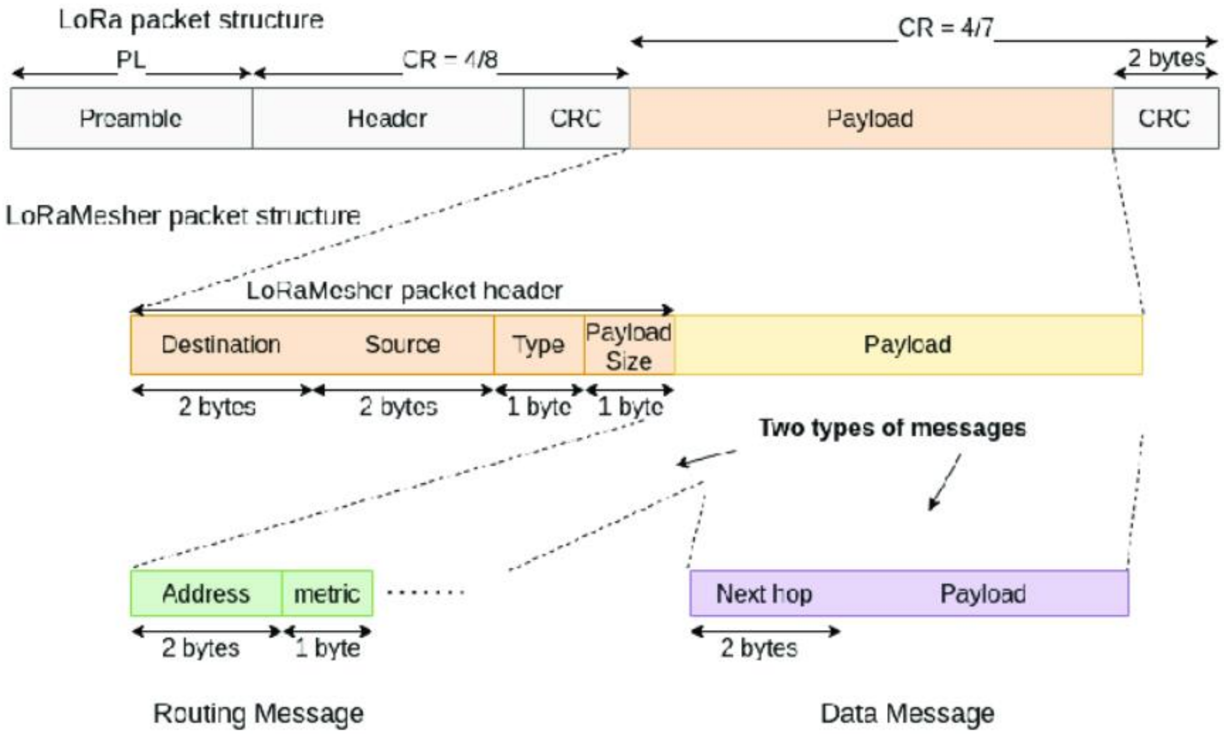

Figure 7: Packet LoRa structure

This figure explains the structure of a LoRa packet and the use of every parameter and frequency along the transmission of this packet.

2.4 Choice analysis:

One of the important questions here is why choose LoRa. There is many reasons to the choice for starters LoRa technology is a key feature of our anti-theft system, which provide a perfect fit for our project. LoRa transceivers have the propensity of extremely low power consumption, so they can function for a significant long time on a battery charge making constantly replace the battery unnecessary. It is an inevitable factor for the system that focuses on the eternal function. With LoRa technology, we are able to transmit data up to 10 kilometers in the countryside and it is perfect for our gadget anti-theft system, which can detect any suspicious activities occurring on the gadgets even in the remote areas. Besides that, LoRa's deep indoor penetrating capability is helpful for ensuring optimal quality of communication on multi-floor buildings which is an indispensable factor in indoor tracking of lost devices. LoRa communicating without protected spectrum licenses helps in removing the expensive frequency license, reducing the deployment expenses for our anti-theft system. While LoRa is able to manage a high traffic, it provides our anti-theft system with the ability to handle large data volume from multiple devices, thus increasing scalability. The overall security features of LoRa protocol, such as unique encryption keys and message acknowledgement make our anti-theft system immune to illegal access and data breaches. However, LoRa's infrastructure cost is also low and has open-source software choices as well which makes LoRa a good system in the context of our anti-theft. Broad LoRa ecosystem, which includes device manufacturers as well as the app developers, is a reliable and robust IoT solution for anti-theft system that we provide. Compare to other IoT communication protocols LoRa stands out when it comes to IoT application as it is compared to cellular technologies like 4G or 5G, LoRa offers significantly longer range and lower power consumption, making it ideal for applications where devices need to operate for extended periods without frequent battery replacements. Additionally, LoRa operates in license-free spectrum, eliminating the need for costly frequency licenses, unlike cellular technologies. In comparison to other low power, wide-area network (LPWAN) technologies like Sigfox, LoRa offers greater flexibility and scalability. LoRa networks can be easily deployed and managed, supporting both public and private network deployments using the same hardware and software. This flexibility allows for easy customization and integration into existing infrastructure. When compared to short-range technologies like Bluetooth and Zigbee, LoRa's long-range capability sets it apart, enabling communication over distances of up to several kilometers in urban environments and even longer ranges in rural areas. This extended range makes LoRa suitable for applications that require connectivity over large areas or in remote locations. Overall, LoRa's combination of long-range communication, low power consumption, and cost-effectiveness make it a compelling choice for a wide range of IoT applications, particularly those that require reliable connectivity over long distances with minimal infrastructure requirements. In figure 6 below, a representation compares LoRa to other protocols using the range and the bandwidth:

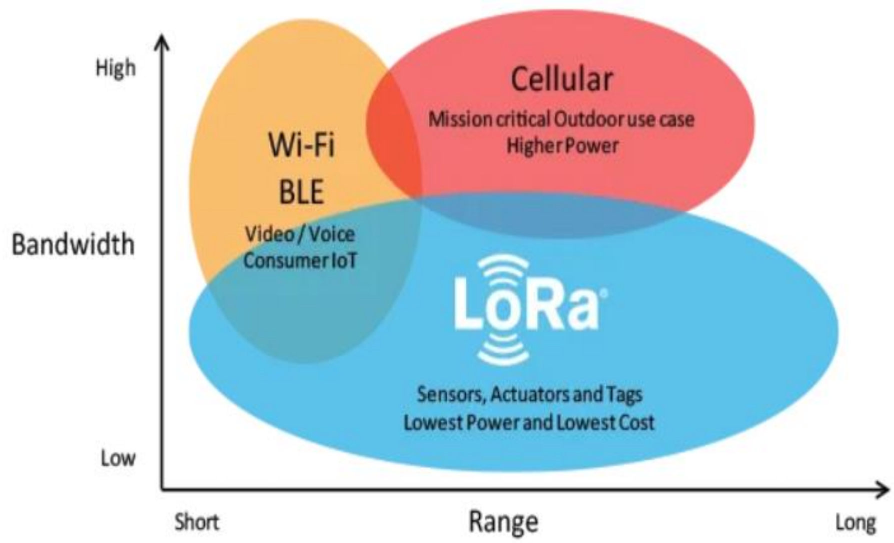

Figure 8: Comparison of LoRa

3. I2C protocol:

3.1 What is I2C? :

I2C, known also as Inter-Integrated Circuit, is short for serial communication I2C is a bus interface connection protocol to access information in an integrated circuit. Phillips Semiconductor initially created this technology, I2C only uses two wires to transmit data between devices (SCL and SDA) and it has become highly popular method of short-range communication.

3.2 I2C functioning:

I2C uses two signal lines: SD and SCLK, which symbolize the data line and the clock line. Initiation of communication opens with the master device sending out a command headed by the address of the slave device that it intends to communicate with. The best-suited slave unit to the addressed ID then replies with an acknowledgment (Ack). The signal, which the SCL line sends on to the master, generates clock pulses needed for data synchronization. Therefore, the data is transmitted between the master and slave machines. Because of that, a bit further in a data flow is set aside for an acknowledgement bit that makes sure a data is received correctly. Once all the information has been shared, the main server gives a stop signal to end this contact. Thus, the devices can be working on different folders since the greater number of devices is wirelessly connected to the one host controller, which is a master and manages the clock signal. Such design is straightforward since I2C protocol just uses just bi-directional communication lines with one line serving as address line and the other data line on which both Master and

Slave parts can listen to or send data. This is why many designers opt to use I2C protocol for connecting various peripherals in embedded systems. In figure7 below, we have the functioning of I2C:

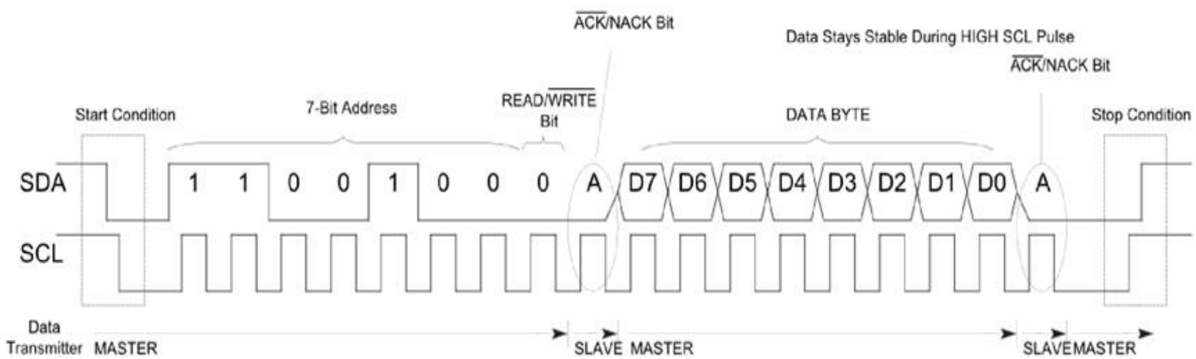

Figure 9: I2C Functioning

The form of the message when it is send via I2C:

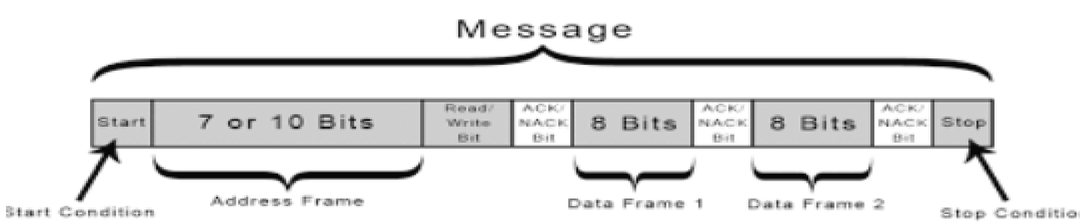

Figure 10: I2C Message format

3.3 Choice analysis:

After thorough consideration, the I2C protocol was selected for facilitating communication between the microcontroller and accelerometer within our anti-theft system. Among the protocols evaluated, including SPI and UART, I2C emerged as the most suitable due to its balanced combination of simplicity, flexibility, and efficiency. Its ability to support multi-device communication on a single bus, along with its straightforward implementation and compatibility with commonly used microcontrollers, makes it an ideal choice for our application. In summary, the I2C protocol was chosen for its robust and reliable communication capabilities, meeting the specific requirements of our anti-theft system. Here we have a table that compares between the three choices considered in this project:

Serial Protocol	UART	I2C	SPI
Protocol Complexity	Simple	Not complex even while chaining multiple devices	It's complex as you increase the number of devices
Speed Rating	Slowest	Medium speed (Faster than UART)	Fastest protocol
No. of Devices	Maximum- 2 devices	It can take up to 127 devices. But, it becomes complex with the addition of devices.	Can take numerous devices but it becomes complex.
No. of Wires	One	Two	Four
Duplex Form	Full Duplex	Half Duplex	Full Duplex
Number of Master Inputs and Slave Inputs	Single to Single	Multiple slaves and masters	One master and multiple slaves.

Table 1: Comparison between the protocols

In conclusion, I2C was the best choice out of the three for data acquisition in out anti-theft system, which will be used between the accelerometer and the MCU we have.

4. STM Nucleo_F446RE:

4.1 What is STM Nucleo_F446RE? :

The STM Nucleo-F446RE is another development board being made by STMicroelectronics based on the STM32F446RE microcontroller. It has the Nucleo-64 family and has the Arduino Uno R3 compatible layout, meaning that is useful for many compatibility boards. The STM32F446RE microcontroller is ARM based, has Core M4 processors in addition to the Nucleo-F446RE board, the ST-LINK/V2-1 offers the possibilities of connecting and programming it.

Figure 11: STM32 development board

Here we have a table that describes the STM32 board:

Feature	**Description**
Microcontroller	STM32F446RE ARM Cortex-M4
Core	ARM 32-bit Cortex-M4
Clock speed	Up to 180 MHz
Flash memory	512 KB
SRAM	128 KB
Operating Voltage	1.7V to 3.6V
Debugging	On board ST-LINK/V2-1 debugger/programmer
Digital I/O Pins	51
Analog Pins	12-bit ADC with up to 16 channels

Table 2: STM32-NucleoF446RE Description

4.2 Choice analysis:

Regarding our project, it was kind of an unanimous decision that we shall utilize an STM Nucleo-F446RE board integrated with a ST-LINK/V2-1 programmer. Unlike other solutions may his describe, our project solely designed and depends on a single little device require no complication in the selection of hardware. In the following segment, he will explore the characteristics of our project in details including what differentiates our solution form others and makes it innovative and practical.

- **The ST-LINK/V2**: is an in-circuit debugger and programmer for the STM32 microcontrollers. The single-wire interface module (SWIM) and JTAG/serial wire debugging (SWD) interfaces are used to communicate with any STM32 microcontroller located on an application board.

5. The LoRa Device:

5.1 What is the LoRa device? :

The key item of our product is a custom wireless device that is suitable for such a task. Be Wireless Solutions team designed and developed the device rigorously ensuring that it satisfies our project's needs. The device has on-board STM32L0xx core, which ensures that the processing is robust and energy efficient in terms of sensor data monitoring and processing services provision, both of which are essential for prolonged operation of the device. At its heart of the device there is a module Cambro-mazantly realized in type ABZ (CMWX1ZZABZ-078) that enables LoRaWAN Long-range Low-problem communication necessary for our deployment scenario. This module is collaborated with SX1276 transceiver. This combination provides the robustness for the reliable data transmission even in the most challenging situations. The LIS2DW12 integrated accelerometer is one of the features in our device that makes it extremely multi-functional and multitasking, and it is an excellent choice for motion detection applications. The creation of interactive use cases improves the chance such as motion-triggered activities or simply the ability to track activities. Fig. 10 below shows the physical design of our device, which is compact and ergonomic.

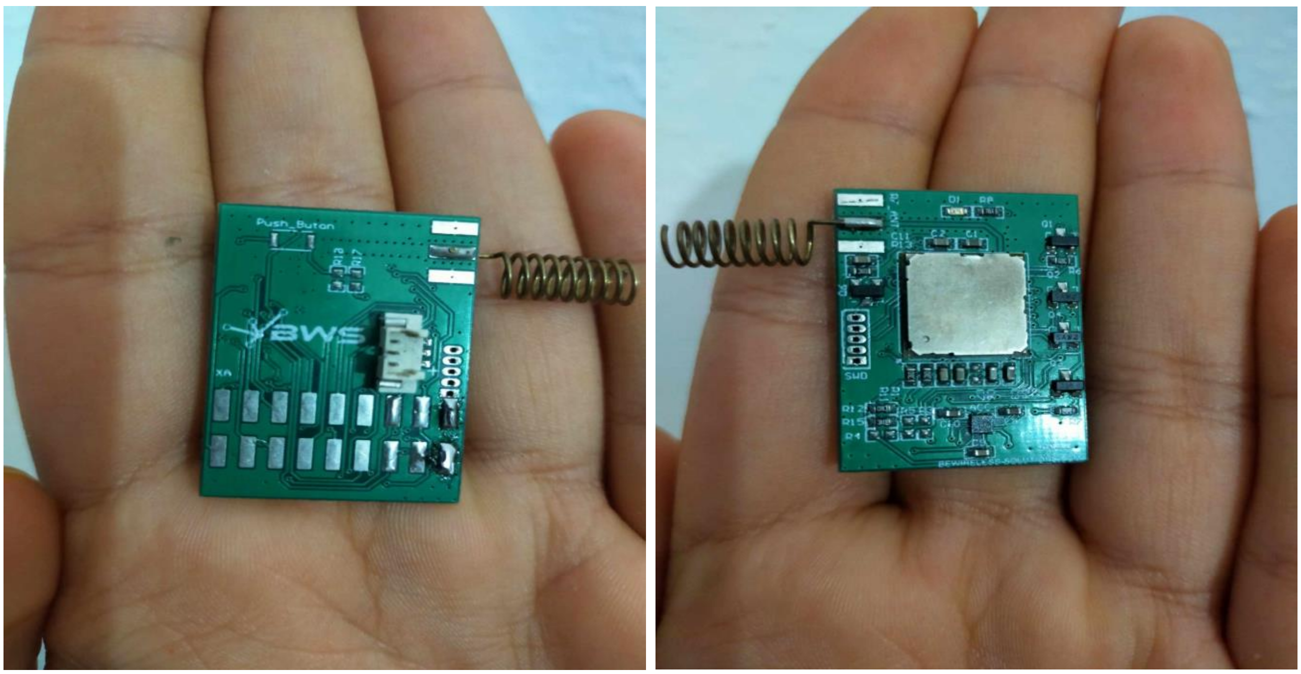

Figure 12: The LoRa device

5.2 Components:

In this part, we are going to talk about each component of the device separately:

5.2.1 STM32L:

Among the STMicroelectronics STM32L series microcontrollers, we feature the devices built for the most extreme energy-saving applications using either ARM Cortex-M0+ or Cortex-M3 processor cores in our case we are using the ARM Cortex-M0+. These include form factor like small, hand-held or wearable; battery power consumption and operation; feature like analog and digital peripherals. The STM32L product range is open to different cases of usage such as IoT, wearables and powered down embedded systems. In the figure 11 below, here is a small description of the stm32L core:

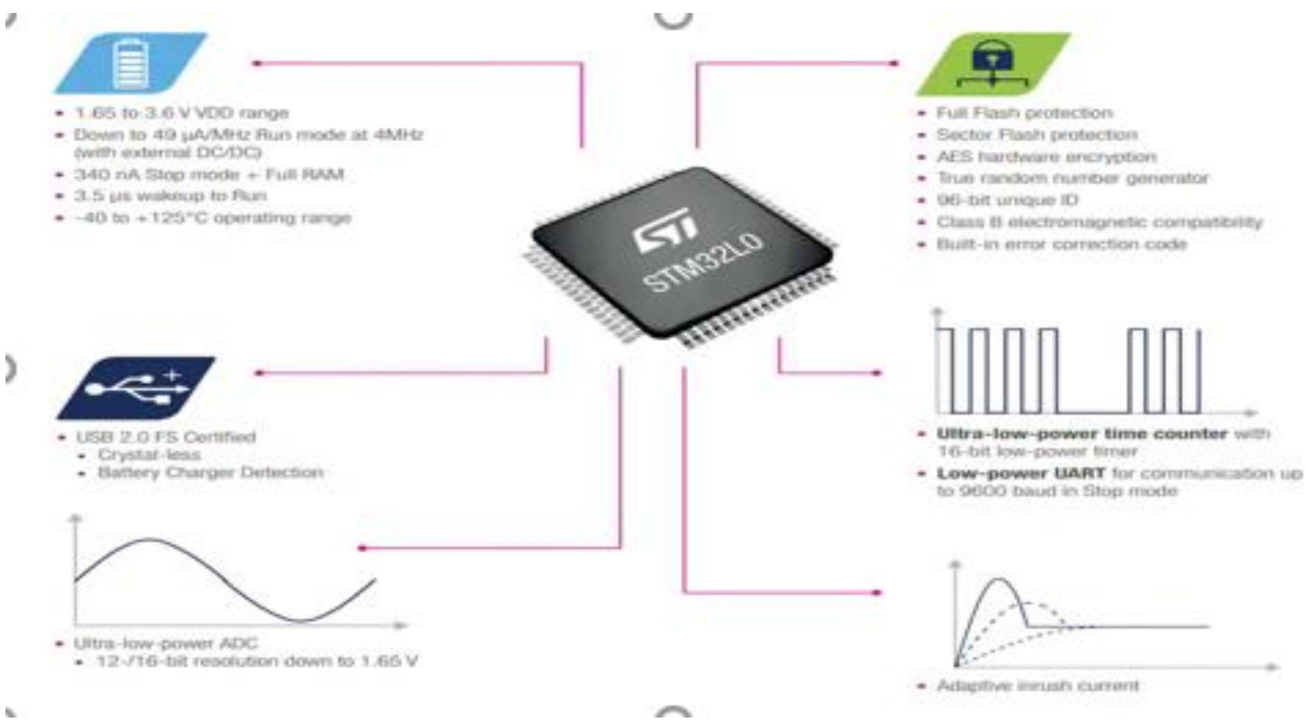

Figure 13: STM32L description

5.2.2 LoRaWAN Module:

The ABZ module is particularly well suited for IoT and M2M applications that require long-range communication and low power consumption. It supports the LoRaWAN protocol, making it compatible with existing LoRaWAN networks, and offers a range of interfaces for easy integration with various microcontrollers and sensors. Overall, the ABZ LoRa module is a versatile and reliable solution for applications that require long-range communication with low power consumption, making it ideal for our project. Here in the figure 12 we have a description of the core of our main device:

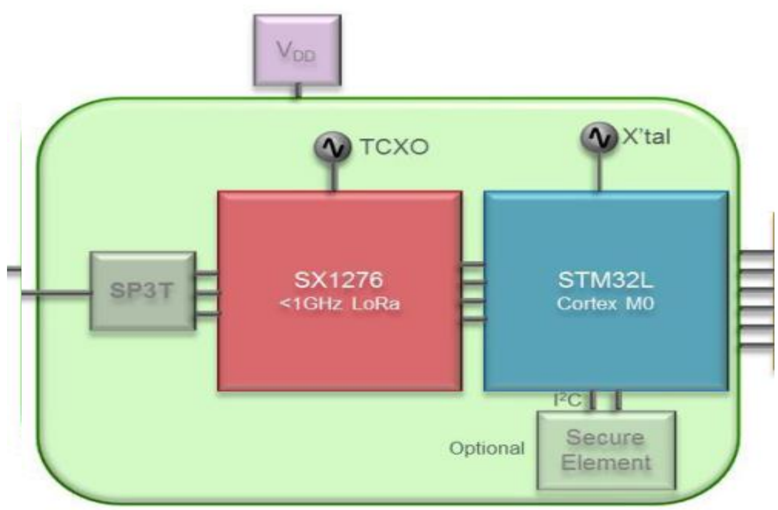

Figure 14: Device core architecture

5.2.3 SX1276 Transceiver:

The SX1276 offers a range of features, including high sensitivity (-148 dBm), adjustable output power up to +20 dBm, and a low power consumption profile, making it suitable for battery-powered devices. It also supports various modulation schemes, including FSK, GFSK, MSK, GMSK, LoRa, and OOK, providing flexibility for different communication needs. In our project, the SX1276 transceiver is integrated into our wireless device, enabling it to communicate with LoRaWAN networks or other compatible devices over long distances with minimal power consumption.

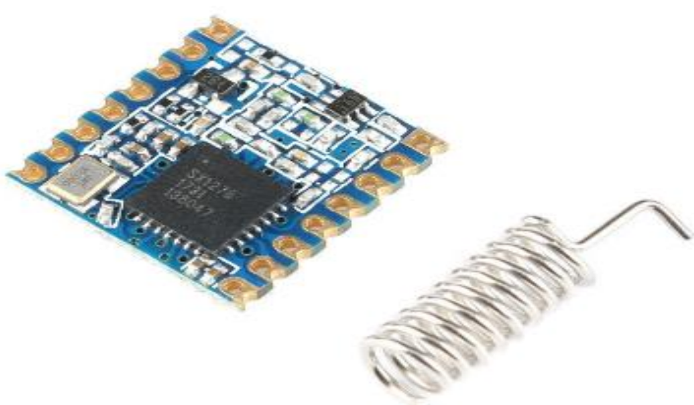

Figure 15: SX1276 Transceiver

5.2.4 The accelerometer:

Key features of the LIS2DW12 include a wide dynamic range (±2g/±4g/±8g/±16g), low noise, and low power consumption. It also includes advanced power-saving features, such as a low-power mode and an embedded FIFO buffer, which help reduce overall system power consumption. LIS2DW12, which has three axes (X, Y, Z), using a MEMS structure to sense acceleration changes. At this moment the MEMS is accelerated and MEMS structure moves down along the direction of change of the electric field. A result is a capacitance changed. Output amounts remaining in digital values are measured by internal circuitry through this modification. The Accelerometer comes along with an integrated DSP capable of data processing and can operate in multi-modes that can be either effective or precise. The error measurements of inaccuracies for our sensors are as follows:

- Zero-g Level Error: Typically ±30 mg (milli-g) over the full temperature range.

- Sensitivity Error: Typically ±3% over the full temperature range.
- Offset Error: Typically ±50 mg over the full temperature range.
- Non-linearity: Typically ±0.1% of full-scale range

5.3 Choice analysis:

- **STM32L:** The choice of the STM32L series microcontroller was driven by its ultra-low-power capabilities, which are crucial for our battery-powered wireless device. The STM32L offers a balance of performance and energy efficiency, making it ideal for our application. Additionally, the STM32L provides a wide range of peripherals and features, ensuring compatibility with our project requirements and facilitating ease of development. Overall, the STM32L was the optimal choice for our project, meeting our criteria for power efficiency, performance, and versatility.

- **LoRaWAN Module:** The choice of the LoRaWAN module type ABZ (CMWX1ZZABZ-078) was based on its compact size, energy efficiency, and integration of the SX1276 transceiver. Compared to other LoRa modules, the ABZ module offered a good balance of performance and power consumption, making it suitable for our battery-powered device. Additionally, the ABZ module's support for the LoRaWAN protocol and various interfaces made it easy to integrate into our system. Compared to other LoRa modules, the ABZ module stood out for its reliability, range, and compatibility with existing LoRaWAN networks.

- **The SX1276 Transceiver:** The significant aspect in this choice is the SX1267 transceiver that has an excellent range and low power consume, very important factors for our wireless devices. To the other comparative devices, transceivers SX1276 are a good match in these parameters – sensitivity, output power and optimization of the power consumption, because of this it was chosen for this task.

- **The accelerometer:** While the preference for LIS2DW12 accelerometer mainly depends on its low power input, wide dynamic range, and versatile functionalities, we finally go with this sensor because it is the only one meeting all the requirements for our wireless device. The LIS2DW12 from LIS3DH clearly hand the better battery consumption and performance for our application, an important factor due to already limited resources. On the other side, the LIS2DW12's internal FIFO buffer and low-power modes were impressive, that ensured power consumption minimization. Unlike the other accelerometers, the LIS2DW12 belong to the best features group, thus make it the best choice for our project motion sensing needs, and clarifies the passage.

IV. Conception UML:

1. Design methodology:

With the conceptualization phase of the project, I made a decision to use the Unified Modeling Language (UML) as the method of developing the system. UML is the standardized way of picturing the design of a system, which guarantees that the team speaks the same language, leading to the progress of understanding between all stakeholders. With UML, I plan to develop all of the descriptions especially use case diagrams, sequence diagrams, activity diagrams, and component diagrams. These charts will not only be a guide during development but also a tool to help us ensure that the product will satisfy the requirements and expectations of the organization’s stakeholders. UML's versatility and adaptability are the main reasons for its application for modeling the intricate interactions and activities of the component parts of the system, which in turn ensure the effectiveness and lessen the inefficiencies in the system.

2. Case Study:

As the final year's project, I am going to propose a new and innovative anti-theft solution targeted to the particular requirements of BWS's sensor protection. In this setup, an accelerometer is built into a LoRa device which has an STM32L core and which communicates with the sensor using I2C. The main objective for this system is to identify and stop movements without permission of the sensors, which means a possible stealing. The system operates by using a quite complicated theft detection algorithm that will be very carefully developed and fine-tuned so that it can be able to detect suspicious activities accurately. When the algorithm is triggered, it will analyze the real time data from the accelerometers, and classify the movement to see if it is part of the usual theft activities. Once the suspicious act is detected, the system will immediately trigger an alert. The alert will be transmitted LoRa to the designated communication gateway immediately which will subsequently be used by the security personnel to respond swiftly to the threat. Moreover, the system will be designed to be scalable and adaptable enough to interface with different types of sensors and different environmental conditions without difficulty. With this high-end anti-theft system, BWS will significantly boost their security measures meant to protect their so-called sensors. This project, which features the practical application of the latest technologies, demonstrates both the new solutions for current problems and my dedication to doing so.

3. Diagrams:

3.1 use case diagram:

In the next part, we illustrate use case diagram of proposed anti-theft hardware. The use case diagram shows the interactions between the users and the system at a high-level. It pinpoints the functionalities of the system. It marks the roles of all the parties played, and the different uses for every party. The diagram is very important to discover the extent of system's applications and will serve as a starting point for descriptions of the system and development processes.

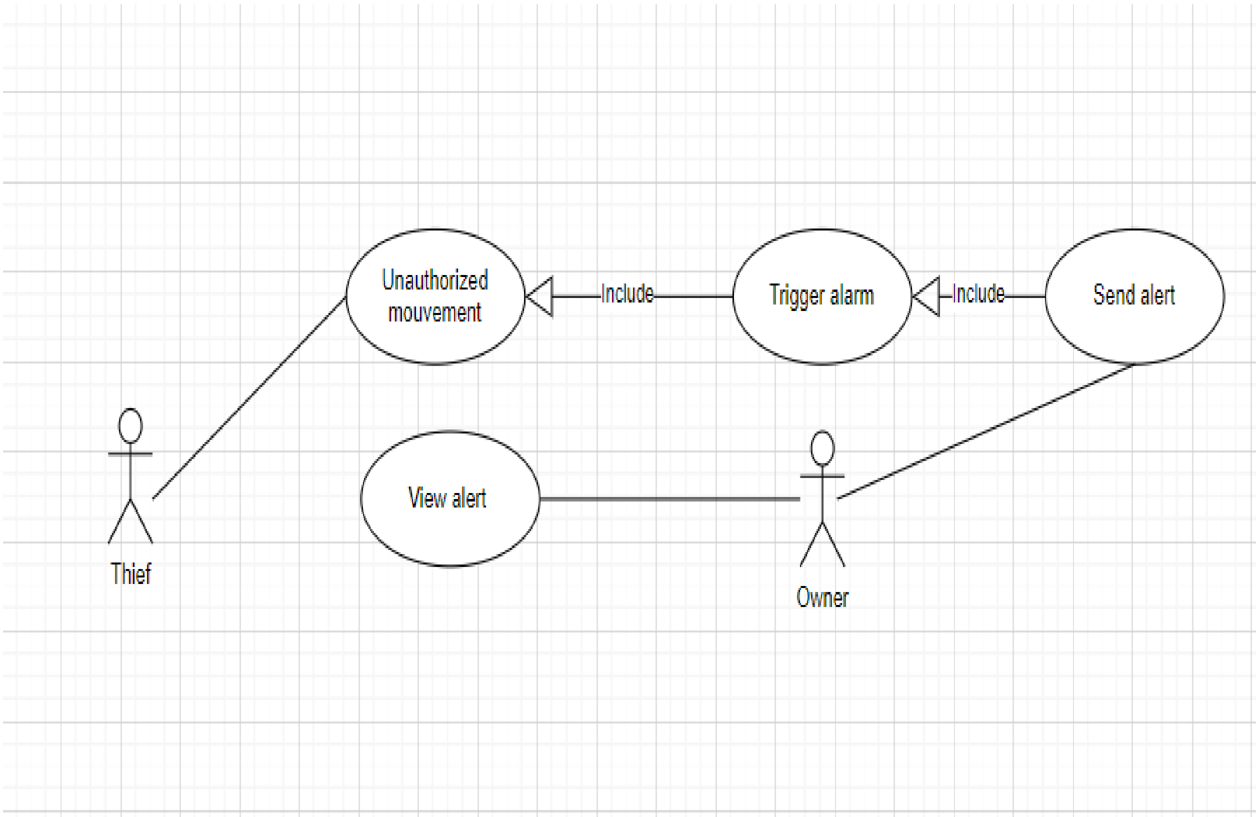

Figure 16: Use case diagram

3.2 Sequence diagram:

In this section, we introduce the pictogram of the anti-theft solution we have prepared. The described diagram shows the flow of interactions between the system objects, or when objects interact in order to perform particular use case. It reveals the process of the sequence, and the sequence of the operations, reflecting how the all the parts collaborate to make the normally operating system to perform the expected functionality. This scheme is the key for explaining the dynamic way in which system is run and for understanding what impact human actions have on environment.

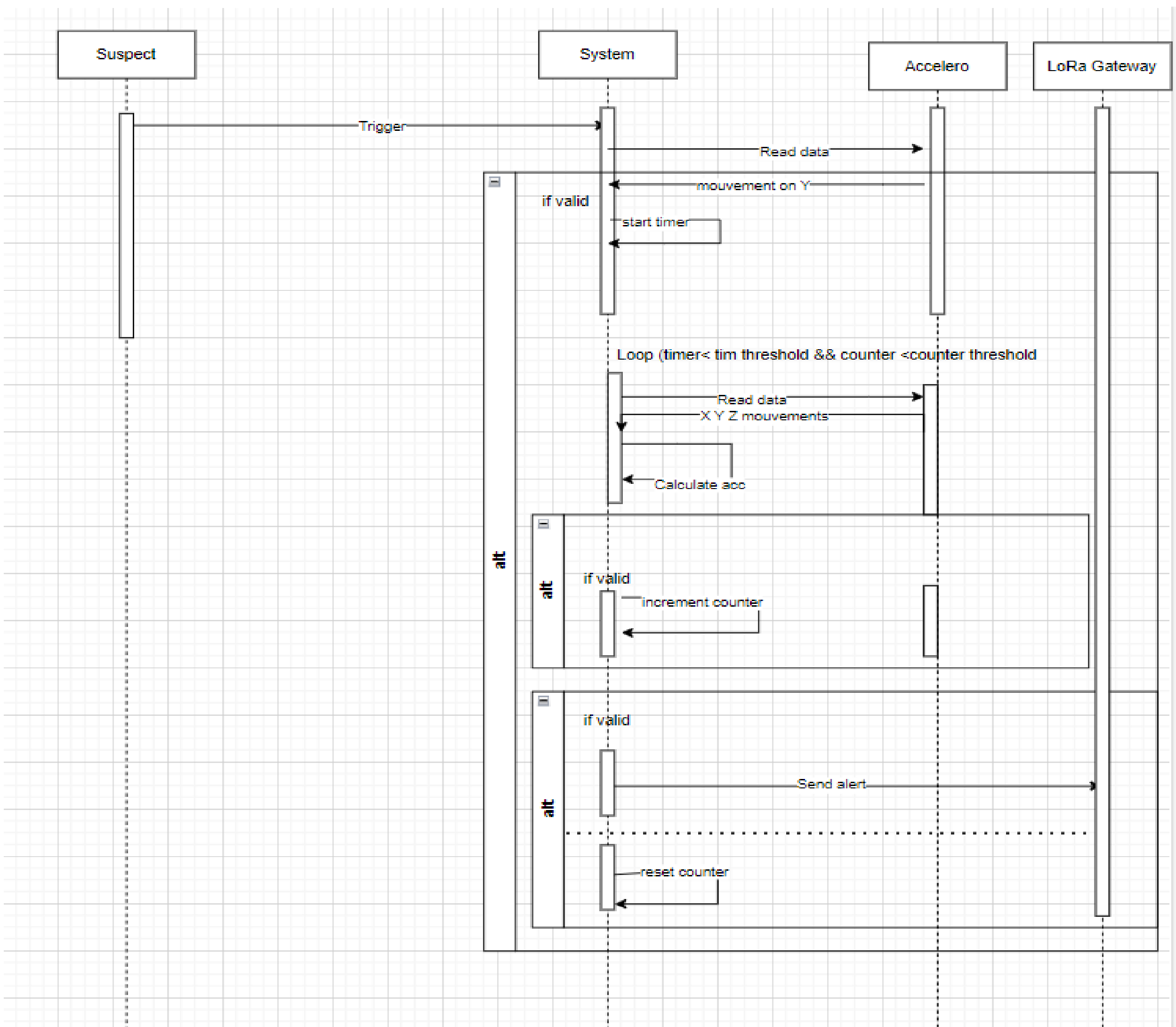

Figure 17: sequence diagram

3.3 Activity diagram:

On this part, we draw an activity graph for our theft systems. The activity diagram is the diagrammatic representation of flows form one activity to another action. They can be shown in sequences, concurrent steps or various processes. It gives a detailed demonstration of the operation, which clearly shows different decision points, concurrent flow paths, and synchronization points of tasks. This flow chart is one of the tools that are significant in the enlightening operational methods and was of great essence in identifying the process improvement in the system.

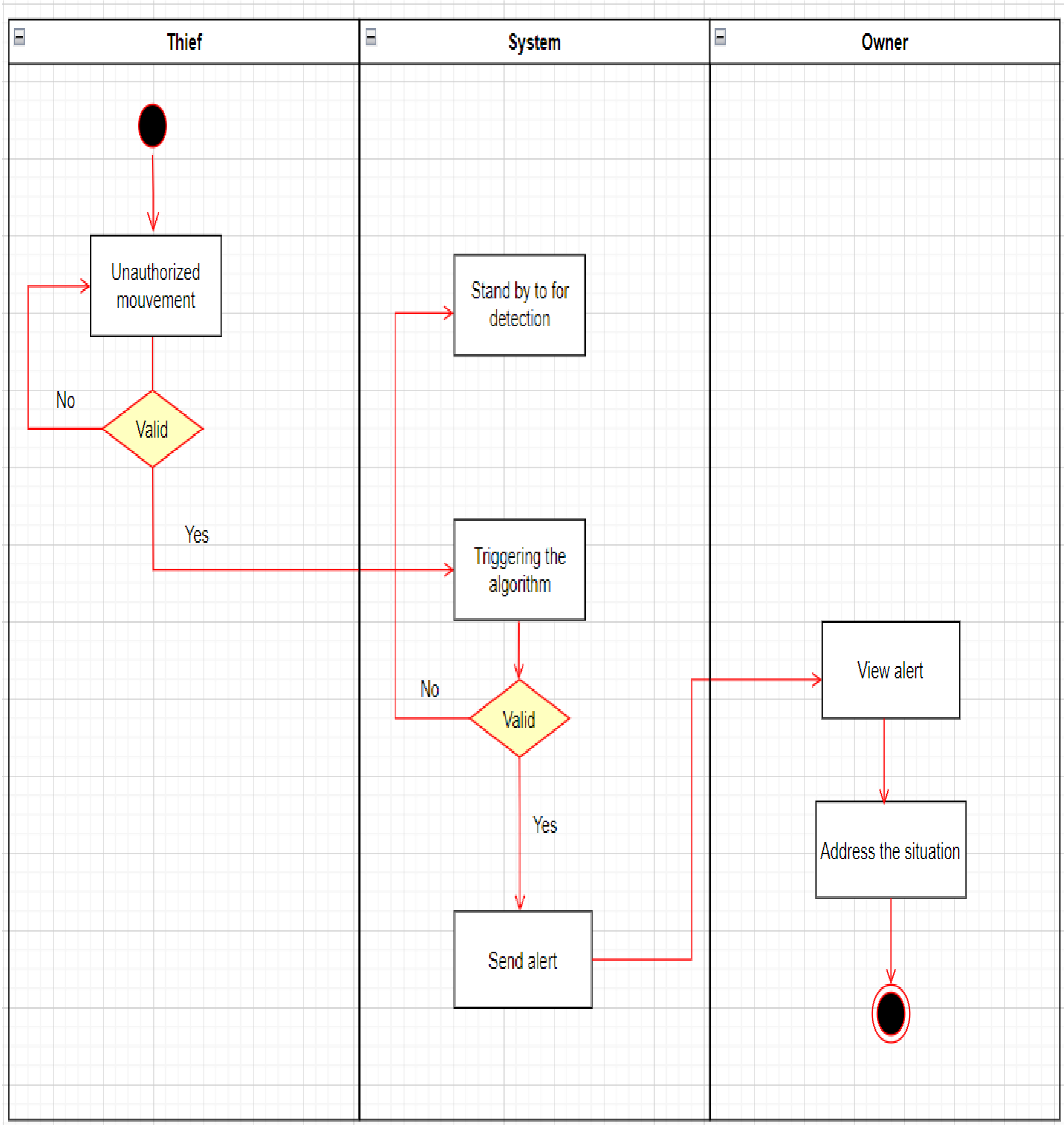

Figure 18: Activity Diagram

V. Conclusion:

In conclusion, Chapter 2 provided an overview of the key components of our theft prevention system. The main hardware and software elements include the STM32L0xx microcontroller, the CMWX1ZZABZ-078 LoRaWAN module, the SX1276 transceiver, and the LIS2DW12 accelerometer. Each component was selected for its low power efficiency, communication range, and motion sensitivity to meet application guidelines. The selection process involved evaluating existing technologies and comparing their performance, cost, and compatibility. We used various diagrams, including use case, sequence, and activity diagrams, to illustrate the system's architecture and interactions. These diagrams laid the groundwork for the system's design and identified essential interactions. Chapter 3 will address the realization phase, detailing the practical implementation and development of the project. This includes hardware assembly, software development, and integration testing, bridging the theoretical foundation of this chapter.

Chapter 3: Realization

I. Introduction:

In this chapter, specifics of how to implement our plan and make the project a success will be outlined. There is a clear, systematic process, which describes the process from the initial idea to the final product, and it is split into three stages. Every phase comprises of significant activities and processes that were important to accomplish the goals of a project. We will first briefly mention the tools and Integrated Development Environments that were used in the development stage. This should involve identification of all software and hardware tools used in the project alongside justifications of their relevance and significance. It is important to consider this choice and usage as a basis for developing the subsequent phases of realization. After the discussion of the tools and IDEs, we will distil the realization into three vital phases. Each phase will show the actions that were completed, what problems were found, and how those were solved to give a better understanding of the project from the beginning until the end of its life cycle. At the end of this chapter, you will have a good understanding on how the project was deliberately and systematically delivered.

II. Working environment :

1. Software:

- **IAR Embedded Workbench**

The Integrated Development Environment that we are going to incorporate for designing our Microcontroller Unit is the IAR Embedded Workbench, which is a product of IAR Systems. It first came into the market in1983. IAR Embedded Workbench is well suited for designing embedded and IoT projects because all project files are collected in one IDE, and navigation is convenient of course it uses C and C++ programming languages. It also has many features that help in developing and debugging your project, these includes the live watch, and this is used in monitoring variable across the project. Furthermore, the Memory Window tool is used to point at the specific flash memory address and display its contents as well. All these tools were useful in my project and their uses shall be analyzed later on. In this case, at the bottom of the page 'IDE Picture', I will attach the picture of the IDE as illustrated in figure 17.

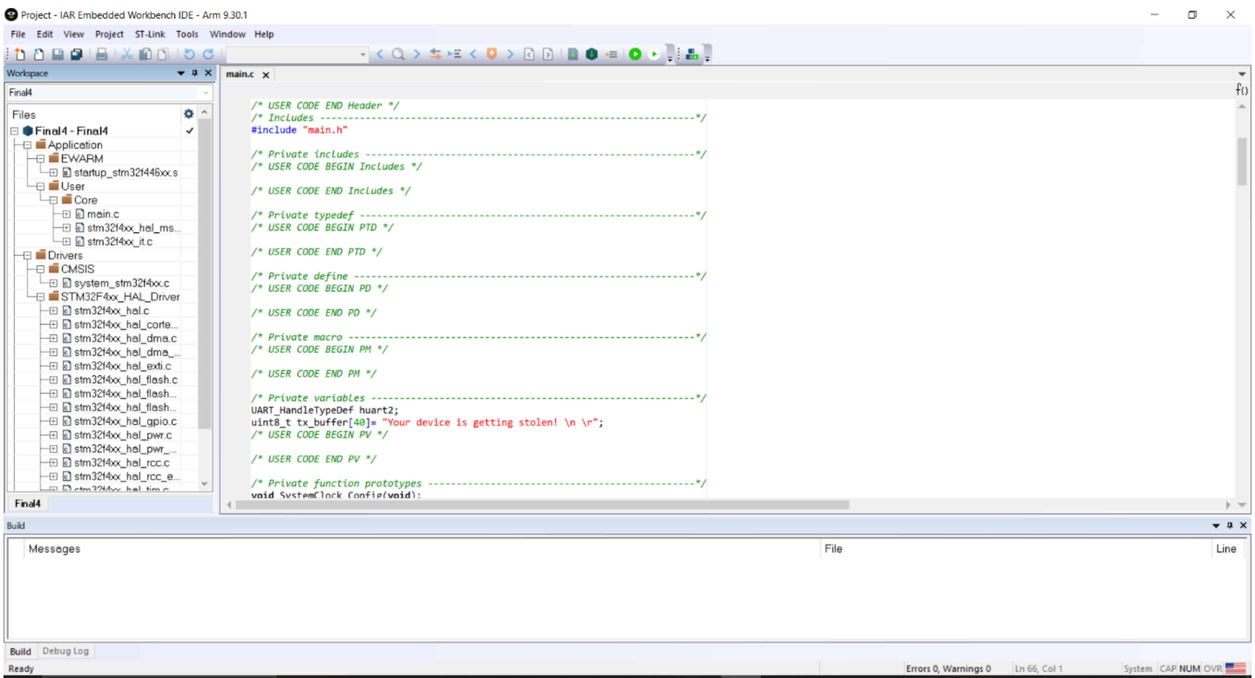

Figure 19: IAR Embedded Workbench

- **STM32CubeMx:**

STM32CubeMX is an open-source tool provided by STMicroelectronics that has been available since 2014. What really makes it very valuable for setting up one's MCU or board is that it supports multiple MCU/Board versions. To persist with STM32CubeMX, you can manage Pins, CLK configurations, and any other required communication we need to implement in our project like UART, I2C, SPI, etc. The tool is quite easy to use, and the framework can create project files for various IDE such as Keil and IAR. Because of this, it is ideal to begin with this guide; it is simple to understand without. As for the visualization of the interface, I will provide the screenshot here.

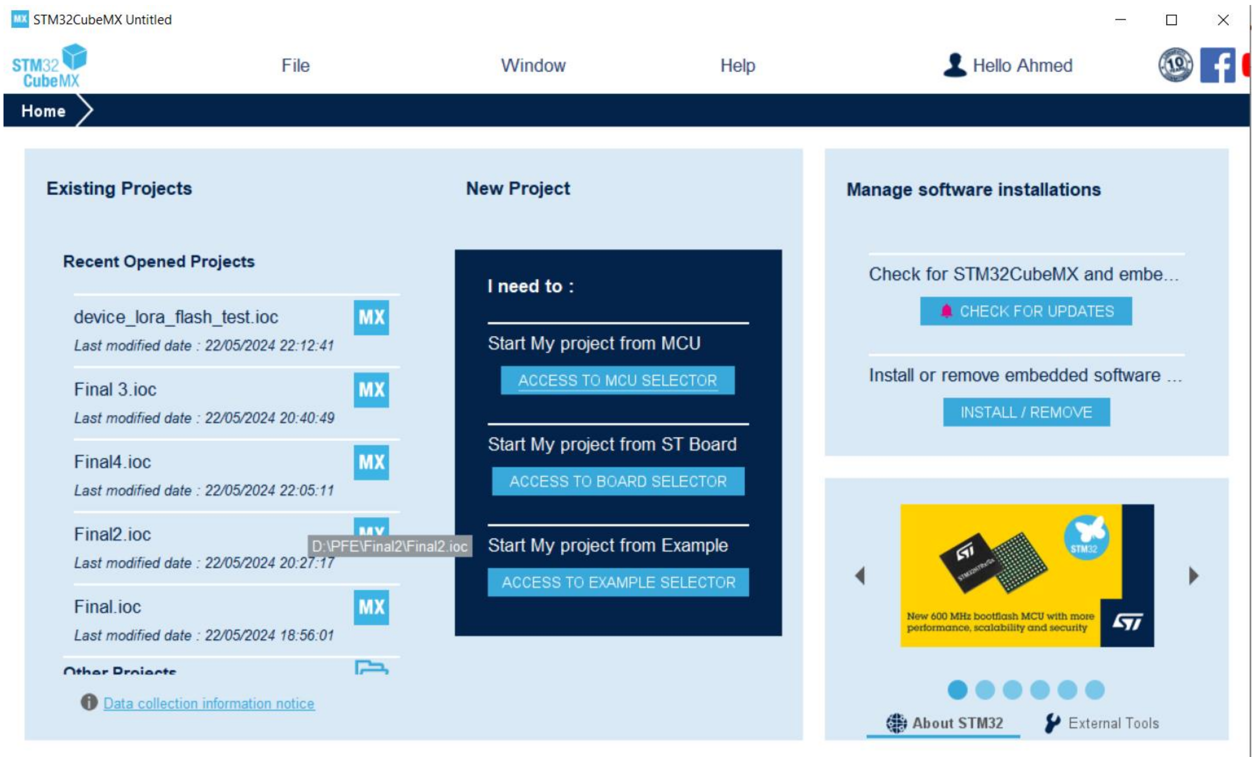

Figure 20: STM32CubeMX Home page

In figure 18, we can see that there is options to open previous projects or start new one through board and MCU selectors also there is examples for beginners to learn about the Tool more.

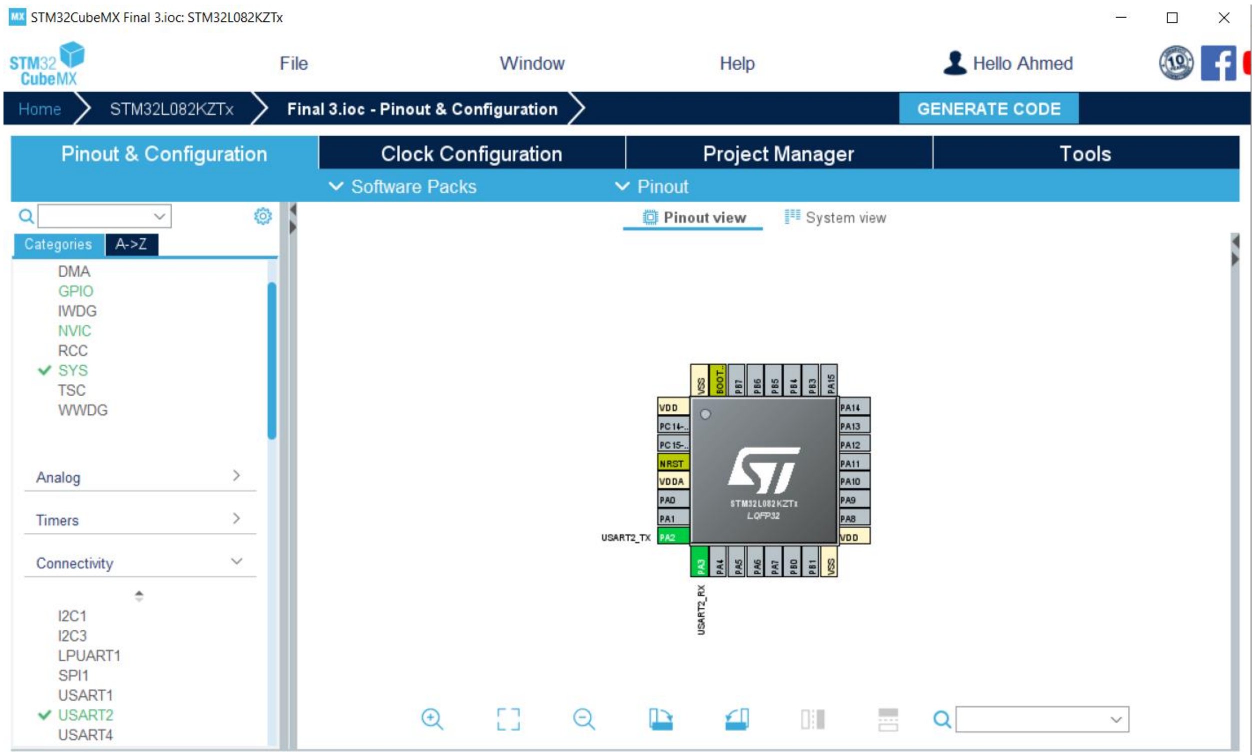

Figure 21: STM32CubeMX Configuration Page

In figure 19, this is the configuration page where we configure the pins, connectivity, and CLK and using the generate button we can generate the project and configuration we set up this tool really saves time.

2. Hardware:

As mentioned in the previous chapter we are going to use:

- LoRa device that has an integrated STM32L0xx core a SX1276 LoRaWAN module and most importantly our lis2DW12 sensor
- ST-Link to program our device

- Second LoRa device just without the accelerometer to use as a LoRa gateway to send the alert to

III. Setting up I2C communication:

1. The process:

First, we applied the STM32CubeMX tool to set up a new project and make devices selection, thus choosing the MCU, the STM32L082KZT. First off, there was not a lot of configuration possible and was only able to get to the clock configuration without any need to configure any pins. The next step was to carefully study the sensor datasheet to better familiarize ourselves with it and determine which registers are required. This was important because it helped a lot when configuring and testing for the actual functionality of the system. From here, we moved on to the generation of the header file for the sensor, namely lis2DW12.h. Accordingly, to the datasheet, since the implementation of the required register is now ready, we need to define what functions are required. Further, actual execution of these functions was done in the lis2DW12 sequence. The

MEMS library was used as the basis for creating and building the c file.

- MEMS, or Micro-Electro-Mechanical Systems, are miniature devices that integrate electrical and mechanical components at a microscopic scale. These systems can include sensors, actuators, and electronic circuits on a single chip. MEMS technology is widely used in various applications, including accelerometers, gyroscopes, pressure sensors, and microfluidics, due to its ability to combine complex functionality with small size and low power consumption.

The source file included the Initialization of the sensor and the initialization of the registers that are needed for the accelerometer to work on X, Y and Z-axis and of course we used +/-2g for the range. Lastly, I used the live watch tool to monitor the changes of the acceleration on each axis using the Live Watch

2. The Challenges:

During this part of the project, we encountered several difficulties, particularly with getting the I2C communication to work correctly. At times, despite the code being correct and successfully downloading and debugging, the device did not function properly. To address this issue, we used a specific register address to test the device's functionality. By sending it a specific hexadecimal value, the device should return another value indicating that the communication is working properly. This method helped us confirm the integrity of the I2C communication. Another challenge we faced was obtaining clear and meaningful values from the sensor. Initially, the sensor provided values that did not make sense. To resolve this, we developed a formula to convert the raw sensor data into logical values expressed in units of m/s^2.

- **Formula: Accel_(X,Y,Z)_RAW / 16384.0*9.80665 ;**

This conversion was essential for ensuring that the sensor data was accurate and usable for our project.

IV. Figuring out an algorithm:

1. The process:

In the beginning, we started thinking about the whole algorithm and possible movements that a thief is going to do while attempting a theft we first added something an algorithm which is thresholds these are acceleration values that are incase surpassed it might possibly be a theft attempt. Then we thought about what if the accelerations or the movements happening on the sensor are not theft attempt what if they are just maintenance or moving the electronic device from one place to another for that we added a counter that counts how many times the threshold is surpassed and one movement that initializes the algorithm in the beginning which is a movement on the Y axis. To test the algorithm and determine the necessary thresholds, we implemented a procedure to store accelerometer values in flash memory. We conducted this by creating a table within the flash memory to log the accelerometer readings. By powering the device with a battery and moving the accelerometer in specific patterns, we collected data to identify the thresholds. Using the IAR Embedded Workbench memory monitor, we examined the values stored at each address in the flash memory. Based on these observations and the average acceleration values typical for human walking and running, we established the following thresholds for our algorithm:

- **Y Threshold =2.0 m/s² :** this is the first threshold used to initialize the first step of the algorithm.
- **Walking Threshold= 1.2 m/s²:** this is the acceleration for a human walking calculated using a formula I will add below.
- **Running Threshold= 9.0 m/²:** this is the acceleration for a human running calculated using the formula.
- **acceleration=**$\sqrt{}(x_{acc}^2 + y_{acc}^2 + z_{acc}^2)$

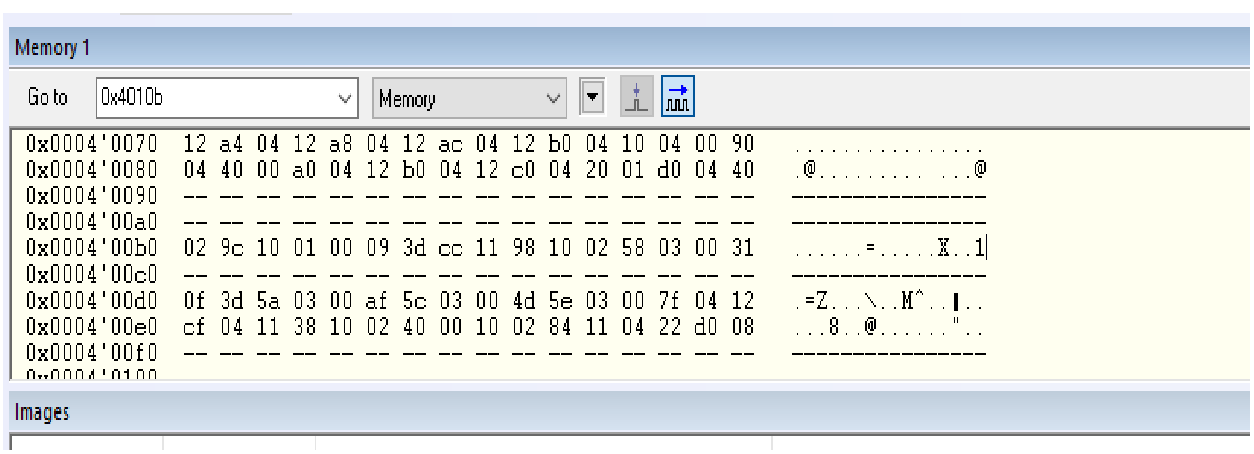

Figure 22: Memory tool in IAR

2. Challenges:

The main difficulties we found in this phase is the way to detect that it is certainly theft that is occurring on the device because when it comes to it detecting human movements specifically theft of course after many algorithms and logical thinking we were able to find a perfect algorithm that uses timers and thresholds. One challenge that is more important was coding the flash memory to be able to add to it a table and save the table in the flash memory so it is not deleted when we remove the power source from the device.

V. Sending the alert and applying FreeRTOS :

1. The process:

1.1. LoRa communication

In this part we started by first understanding our LoRaWAN module which is the SX1276 after that we found a source and header file for our module. Then we modified it to get what we need as the SX1276 has both FSK communication and LoRA communication then we started programming our main device by using the algorithm from before which, means if a theft is occurring that is where our **Send_alert** function starts working. Inside our send alert, we have the initialization of our module, which is going to power up the module, and get it ready to work. Then we set up the module for transmitting using **Set_TxConfig** function, inside it we set up a bunch of parameters so that it sends to our gateway and it is going to send an alert message to the gateway using **SX_Send** function. For our gateway, we did the same thing as earlier just this time we set the second device as a receiver using **Set_RXConfig** function and to get it to listen we used the **Set_RX function**. So that the device and gateway are configured, we set up these parameters:

- **RF frequency= 868000000 Hz:** This is the operating frequency of the radio. Both devices must be set to the same frequency so they can communicate on the same channel.
- **LoRa Bandwidth= 125 Hz:** This is the bandwidth of the LoRa modulation. It determines how wide the signal is and affects the data rate and range.

- **LoRa spreading factor = 7:** This parameter determines the spreading factor used in the LoRa modulation it affects the sensitivity and data rate of the communication.
- **LoRa coding rate= 1:** affects the error correction capability of the system.
- **LoRa Preamble length= 8:** It helps the receiver synchronize with the incoming signal

To test this part of the communication we used an UART (Universal Asynchronous Receiver-Transmitter) communication protocol.

- **UART:** defines a protocol, or set of rules, for exchanging serial data between two devices.

Which by using a USB cable we connected the receiver to the PC and using PUTTY we were able to view the message as it is received, below in figure 21 we have the serial monitor provided by PUTTY.

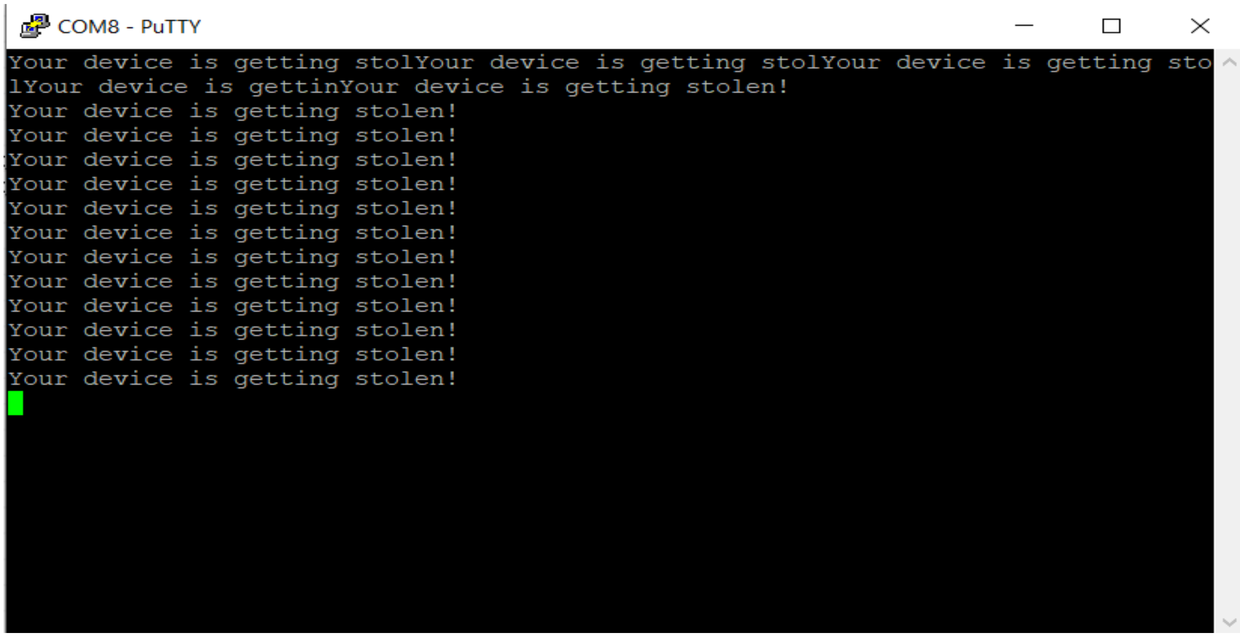

Figure 23: PuTTY Serial monitor

By that we tested our LoRa Communication if it actually works and send a full message. Here we have in figure 22 and explanation of how we tested the communication.

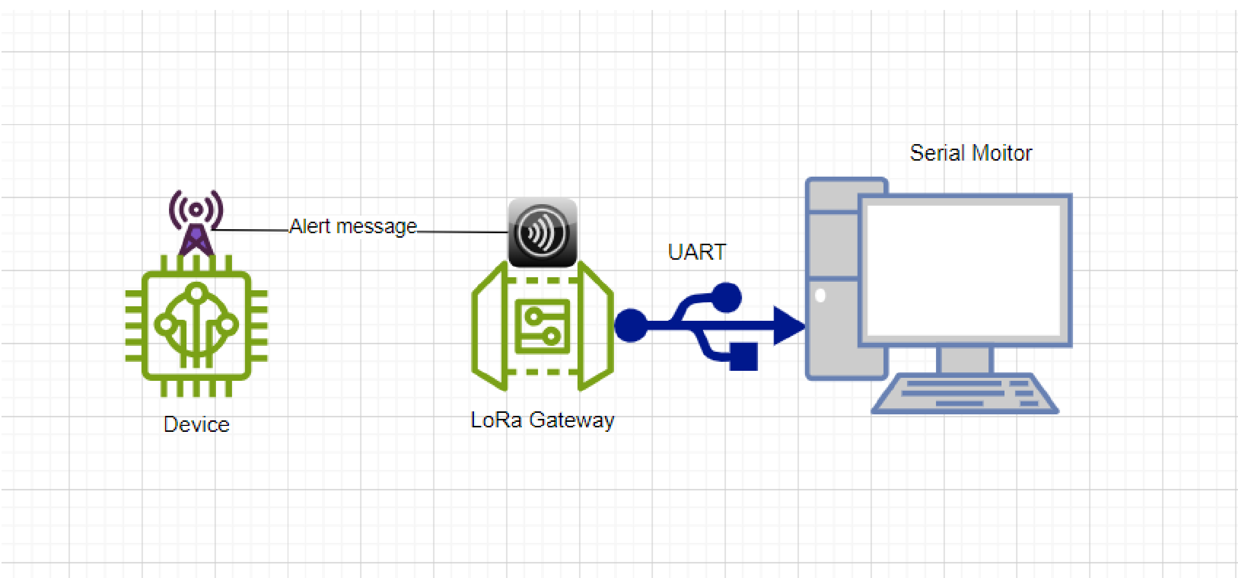

Figure 24: LoRa Communication Test

1.2. Implementing FreeRTOS:

- **FreeRTOS** is an open source, small footprint, real time operating system kernel whose primary goal is to provide efficient scheduling for an embedded system application. Combination of hardware and software, small size, huge processing speed, operation reliability and other beneficial features make it perfect for microcontroller-based projects that require real time performance and multitasking. FreeRTOS controls several tasks so that they can run at the same time while maintaining easy resource computation and task prioritization. It offers tasks, time tools, and synchronization tools and interrupt features that offer a good experience to programmers who often use it to incorporate real-time services into their projects.

In this part, it was not complicated to add FreeRTOS as it is not hard seeing that STM32CubeMX provides the option to add FreeRTOS to your project and add all the tasks we need.

2. Challenges:

While effecting LoRa communication system some of the challenges experienced include; one of the issues to overcome was the problem of setting the correct data rate and bandwidth at which it could achieve the intended range and data transfer rate while using minimal power. Another difficulty that had been encountered was in control and coordination between the transmitter and receiver nodes. Coping with the packet loss and achieving the guaranteed reliable message delivery over the distance was another concern that needs the special attention on the LoRa modulation parameters and the error control techniques.

VI. Result and demonstration:

Now we come to the last part which is assembling everything into one whole project and getting the whole system to work. All we had to do is make a header and source file which there we all include all the libraries needed:

- Lisdw12.h: for the accelerometer
- Sx1276.h: for our transceiver

- Radio.h: for our LoRa initialization

After that we set up the gateway as shown previously with UART communication with the PC, with the help of PuTTY we were able to view our alert message as it is sent to out gateway.

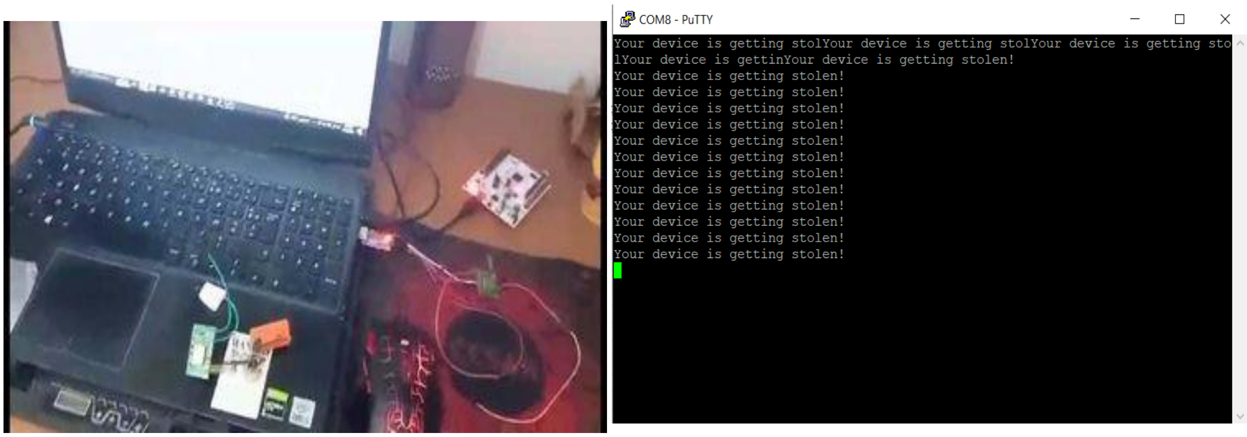

Figure25: Demnstration and result

In figure 25 we can actually see our actual set for demonstration and the result on the serial monitor of PuTTY we attemeted some mouvements and we got the alert then we stopped doing the mouvements the alerts did not send anymore.

VII. Conclusion:

Overall, the realization phase of the project has been successfully implemented and the blueprint creates a strong anti-theft system for sensors in BWS. This was conducted in a controlled working environment for the project where the

STM32 microcontroller and LoRa were used for communicating effectively. Essential factors such as the STM32L0xx core, LoRaWAN module CMWX1ZZABZ-078, SX1276 transceiver, and LIS2DW12 accelerometer were also procured and integrated as per project plan. The software part comprised the STM32CubeMX for selecting and initializing the MCU and generating the associated start-up code and the STM32Cube HAL for the drivers. Coding and debugging of the developed software was done using IAR Embedded Workbench, which gives a surety for the developed software. During the implementation process, there were some issues noted especially when setting the best LoRa communication parameters and when attempting to achieve constant data transmission over very large distance. Nevertheless, these challenges were met and averted through the right adjustment of the parameters and good error control mechanisms. In general, the installation of the anti-theft system can be considered successful, as the entire project objectives have been achieved, and the Hardware and Software sections have shown the efficiency of the selected technologies.

General Conclusion

In conclusion, the implementation of this innovative anti-theft system using IoT technologies has been a significant achievement. By integrating accelerometer data processing and LoRa communication, the system offers a robust and scalable solution for enhancing security in various applications. This project not only meets the specific requirements of BWS but also demonstrates the broader potential of IoT in creating safer environments, aligning with the concept of smart cities. The successful realization of this project underscores the importance of technological advancements in ensuring safety and security in modern urban settings. By leveraging IoT capabilities, the system not only addresses the immediate needs of BWS but also paves the way for future applications in smart cities and beyond. Furthermore, the project's implementation has provided valuable insights into the practical challenges and considerations involved in deploying IoT solutions for security purposes. These insights can serve as a valuable resource for future projects in similar domains, contributing to the advancement of IoT technology in enhancing safety and security measures. In summary, this project represents a significant step forward in leveraging IoT technologies for security applications, with implications that extend beyond the specific requirements of BWS to broader contexts such as smart cities. Its success highlights the potential of IoT in creating safer environments and underscores the importance of technological innovation in addressing contemporary challenges.

Appendix

Here we are going to add some figures and photos of our components.

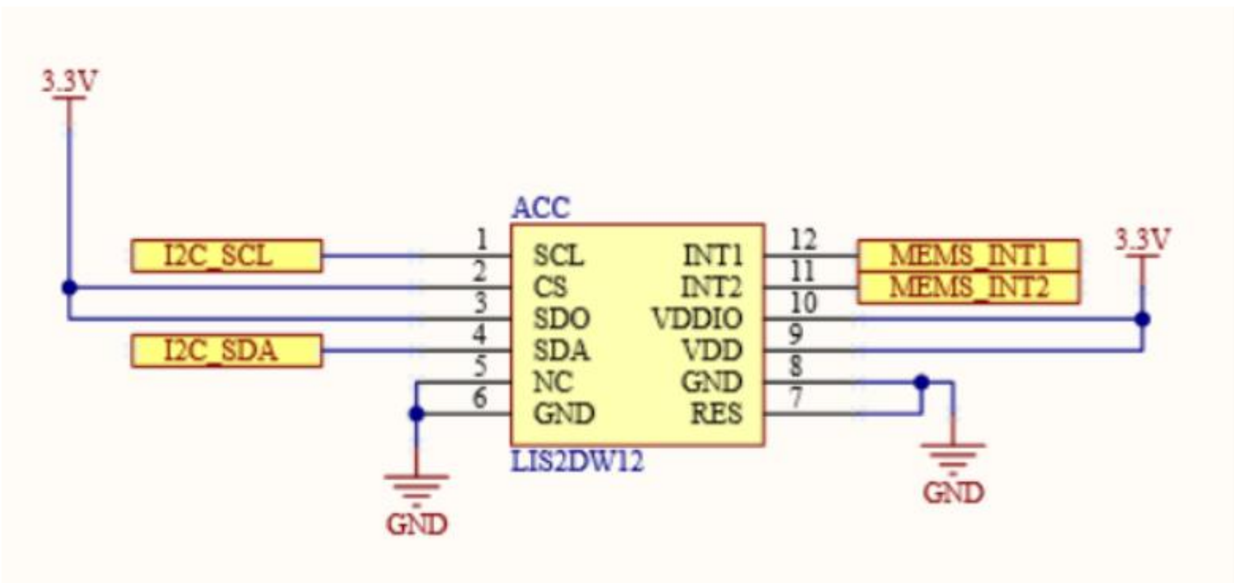

Figure Appendix 1: The accelerometer pins

Figure Appendix 2: I2C Connections

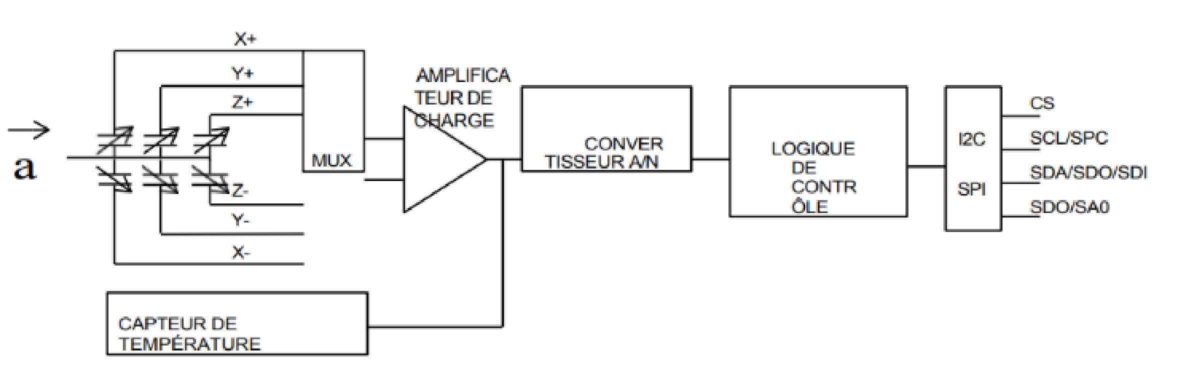

Figure Appendix 3: Architecture de lis2DW12

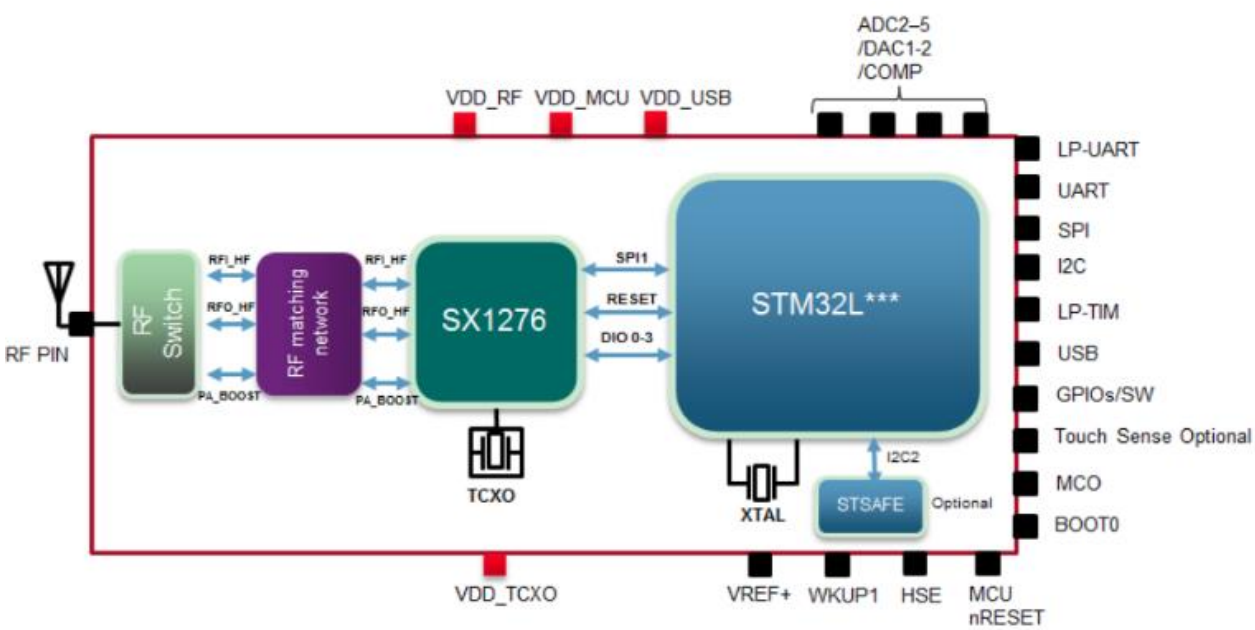

Figure Appendix 4: The main device architecture

References

https://www.techtarget.com/iotagenda/definition/embedded-system

https://www.ibm.com/topics/internet-of-thingshttps://www.geeksforgeeks.org/embedded-c/

https://www.st.com/en/evaluation-tools/nucleo-g070rb.html

https://www.geeksforgeeks.org/i2c-communication-protocol/

https://www.trendmicro.com/vinfo/us/security/definition/lorawan

https://www.deviceauthority.com/blog/unpacking-iot-architecture